SIDROPS
Internet-Draft
Intended status: Standards Track
Expires: 7 June 2026

J. Snijders
BSD
B. Bakker
T. Bruijnzeels
RIPE NCC
T. Buehler
OpenBSD

4 December 2025

A Profile for Resource Public Key Infrastructure (RPKI) Canonical Cache Representation (CCR)
draft-ietf-sidrops-rpki-ccr-02

Abstract

This document specifies a Canonical Cache Representation (CCR) content type for use with the Resource Public Key Infrastructure (RPKI). CCR is a DER-encoded data interchange format which can be used to represent various aspects of the state of a validated cache at a particular point in time. The CCR profile is a compact and versatile format well-suited for a diverse set of applications such as audit trail keeping, validated payload dissemination, and analytics pipelines.

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 https://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 7 June 2026.

Copyright Notice

Table of Contents

1. Introduction

This document specifies a Canonical Cache Representation (CCR) content type for use with the Resource Public Key Infrastructure (RPKI). A validated cache contains all RPKI objects that the Relying Party (RP) has verified to be valid according to the rules for validation (see [RFC6487], [RFC6488], [RFC9286]). CCR is a data interchange format using Distinguished Encoding Rules (DER, [X.690]) which can be used to represent various aspects of the state of a validated cache at a particular point in time. The CCR profile is a compact and versatile format well-suited for a diverse set of applications such as audit record keeping, validated payload dissemination, and analytics pipelines.

The format was primarily designed to support comparative analysis of uniformities and differences among multiple RP instances using different RPKI transport protocols (such as [RFC5781], [RFC8182], and [I-D.ietf-sidrops-rpki-erik-protocol]).

1.1. Requirements Language

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

2. The Canonical Cache Representation content type

The content of a CCR file is an instance of ContentInfo.

The contentType for a CCR is defined as id-ct-rpkiCanonicalCacheRepresentation, with Object Identifier (OID) 1.2.840.113549.1.9.16.1.54.

The content is an instance of RpkiCanonicalCacheRepresentation.

3. The Canonical Cache Representation content

The content of a Canonical Cache Representation is formally defined as follows:

```
RpkiCanonicalCacheRepresentation-2025
  { iso(1) member-body(2) us(840) rsadsi(113549)
    pkcs(1) pkcs9(9) smime(16) mod(0) id-mod-rpkiCCR-2025(TBD) }

DEFINITIONS EXPLICIT TAGS ::=
BEGIN
```

```
   IMPORTS
     CONTENT-TYPE, Digest, DigestAlgorithmIdentifier,
       SubjectKeyIdentifier
     FROM CryptographicMessageSyntax-2010 -- in [RFC6268]
       { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
         pkcs-9(9) smime(16) modules(0) id-mod-cms-2009(58) }

     ASID, ROAIPAddressFamily
     FROM RPKI-ROA-2023 -- in [RFC9582]
       { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
         pkcs9(9) smime(16) mod(0) id-mod-rpkiROA-2023(75) }

     CertificateSerialNumber, SubjectPublicKeyInfo
     FROM PKIX1Explicit-2009
       { iso(1) identified-organization(3) dod(6) internet(1)
         security(5) mechanisms(5) pkix(7) id-mod(0)
         id-mod-pkix1-explicit-02(51) }

     AccessDescription, KeyIdentifier
     FROM PKIX1Implicit-2009
       { iso(1) identified-organization(3) dod(6) internet(1)
         security(5) mechanisms(5) pkix(7) id-mod(0)
         id-mod-pkix1-implicit-02(59) }
     ;

   ContentInfo ::= SEQUENCE {
     contentType      CONTENT-TYPE.&id({ContentSet}),
     content      [0] EXPLICIT CONTENT-TYPE.&Type({ContentSet}{@contentType}) }

   ContentSet CONTENT-TYPE ::= {
     ct-rpkiCanonicalCacheRepresentation, ... }

   ct-rpkiCanonicalCacheRepresentation CONTENT-TYPE ::=
     { TYPE RpkiCanonicalCacheRepresentation
       IDENTIFIED BY id-ct-rpkiCanonicalCacheRepresentation }

   id-ct-rpkiCanonicalCacheRepresentation OBJECT IDENTIFIER ::=
     { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
       pkcs-9(9) id-smime(16) id-ct(1) ccr(54) }

   RpkiCanonicalCacheRepresentation ::= SEQUENCE {
     version     [0] INTEGER DEFAULT 0,
     hashAlg         DigestAlgorithmIdentifier,
     producedAt      GeneralizedTime,
     mfts        [1] ManifestState OPTIONAL,
     vrps        [2] ROAPayloadState OPTIONAL,
     vaps        [3] ASPAPayloadState OPTIONAL,
     tas         [4] TrustAnchorState OPTIONAL,
```

```
    rks         [5] RouterKeyState OPTIONAL,
    ... }
    -- at least one of mfts, vrps, vaps, tas, or rks MUST be present
    ( WITH COMPONENTS { ..., mfts PRESENT } |
      WITH COMPONENTS { ..., vrps PRESENT } |
      WITH COMPONENTS { ..., vaps PRESENT } |
      WITH COMPONENTS { ..., tas PRESENT } |
      WITH COMPONENTS { ..., rks PRESENT } )

  ManifestState ::= SEQUENCE {
    mis               SEQUENCE OF ManifestInstance,
    mostRecentUpdate  GeneralizedTime,
    hash              Digest }

  ManifestInstance ::= SEQUENCE {
    hash              Digest,
    size              INTEGER (1000..MAX),
    aki               KeyIdentifier,
    manifestNumber    INTEGER (0..MAX),
    thisUpdate        GeneralizedTime,
    locations         SEQUENCE SIZE (1..MAX) OF AccessDescription,
    subordinates      SEQUENCE (SIZE(1..MAX)) OF SubjectKeyIdentifier
                        OPTIONAL }

  ROAPayloadState ::= SEQUENCE {
    rps               SEQUENCE OF ROAPayloadSet,
    hash              Digest }

  ROAPayloadSet ::= SEQUENCE {
    asID              ASID,
    ipAddrBlocks      SEQUENCE (SIZE(1..2)) OF ROAIPAddressFamily }

  ASPAPayloadState ::= SEQUENCE {
    aps               SEQUENCE OF ASPAPayloadSet,
    hash              Digest }

  ASPAPayloadSet ::= SEQUENCE {
    customerASID      ASID,
    providers         SEQUENCE (SIZE(1..MAX)) OF ASID }

  TrustAnchorState ::= SEQUENCE {
    skis              SEQUENCE (SIZE(1..MAX)) OF SubjectKeyIdentifier,
    hash              Digest }

  RouterKeyState ::= SEQUENCE {
    rksets            SEQUENCE OF RouterKeySet,
    hash              Digest }
```

```
   RouterKeySet ::= SEQUENCE {
     asID              ASID,
     routerKeys        SEQUENCE (SIZE(1..MAX)) OF RouterKey }

   RouterKey ::= SEQUENCE {
     ski               SubjectKeyIdentifier,
     spki              SubjectPublicKeyInfo }

   END
```

3.1. version

The version field contains the format version for the RpkiCanonicalCacheRepresentation structure, in this version of the specification it MUST be 0.

3.2. hashAlg

The hashAlg field specifies the algorithm used to construct the message digests. This profile uses SHA-256 [SHS], therefore the OID MUST be 2.16.840.1.101.3.4.2.1.

3.3. producedAt

The producedAt field contains a GeneralizedTime and indicates the moment in time the CCR was generated.

3.4. State aspect fields

Each CCR contains one or more fields representing particular aspects of the cache’s state. Implementers should note the ellipsis extension marker in the RpkiCanonicalCacheRepresentation ASN.1 notation and anticipate future changes as new signed object types are standardized.

Each state aspect generally consists of a sequence of details extracted from RPKI Objects of a specific type, along with a digest computed by hashing the aforementioned DER-encoded sequence, optionally including some metadata.

3.4.1. ManifestState

An instance of ManifestState represents the set of valid, current Manifests ([RFC9286]) in the cache. It contains three fields: mis, mostRecentUpdate, and hash.

3.4.1.1. ManifestInstance

The mis field contains a SEQUENCE of ManifestInstance. There is one ManifestInstance for each current manifest. A manifest is nominally current until the time specified in nextUpdate or until a manifest is issued with a greater manifestNumber, whichever comes first (see Section 4.2.1 of [RFC9286]).

A ManifestInstance is a structure consisting of the following fields:

hash the hash of the represented DER-encoded manifest object

size the size of the represented DER-encoded manifest object

aki the manifest issuer's key identifier

manifestNumber the manifest number contained within the manifest's eContent field

thisUpdate the thisUpdate contained within the manifest's eContent field

locations a sequence of AccessDescription instances from the manifest's End-Entity certificate's Subject Information Access extension

subordinates a optional non-empty SEQUENCE of SubjectKeyIdentifier

The subordinates field represents the keypairs associated with the set of non-revoked, non-expired, validly signed, certification authority (CA) resource certificates subordinate to the manifest issuer. Each SubjectKeyIdentifier is the 160-bit SHA-1 hash of the value of the DER-encoded ASN.1 bit string of the resource certificate's Subject Public Key, as described in Section 4.8.2 of [RFC6487]. The sequence elements of the subordinates field MUST be sorted in ascending order by interpreting each SubjectKeyIdentifier value as an unsigned 160-bit integer and MUST be unique with respect to each other.

The sequence elements in the mis field MUST be sorted in ascending order by hash value contained in each instance of ManifestInstance and MUST be unique with respect to the other instances of ManifestInstance.

3.4.1.2. mostRecentUpdate

The mostRecentUpdate is a metadata field which contains the most recent thisUpdate amongst all current manifests represented by the ManifestInstance structures. If the mis field contains an empty sequence, the mostRecentUpdate MUST be set to the POSIX Epoch ("19700101000000Z").

3.4.1.3. hash

The hash field contains a message digest computed using the mis value (encoded in DER format) as input message.

3.4.2. ROAPayloadState

An instance of ROAPayloadState contains a field named rps which represents the current set of Validated ROA Payloads (Section 2 of [RFC6811]) encoded as a SEQUENCE of ROAPayloadSet instances.

The ROAPayloadSet structure is modeled after the RouteOriginAttestation (Section 4 of [RFC9582]). The asID value in each instance of ROAPayloadSet MUST be unique with respect to other instances of ROAPayloadSet. The contents of the ipAddrBlocks field MUST appear in canonical form and ordered as defined in Section 4.3.3 of [RFC9582].

The hash field contains a message digest computed using the rps value (encoded in DER format) as input message.

3.4.3. ASPAPayloadState

An instance of ASPAPayloadState contains an aps field which represents the current set of deduplicated and merged ASPA payloads ([I-D.ietf-sidrops-aspa-profile]) ordered by ascending customerASID value encoded as a SEQUENCE of ASPAPayloadSet instances. The customerASID value in each instance of ASPAPayloadSet MUST be unique with respect to other instances of ASPAPayloadSet.

The ASPAPayloadSet structure is modeled after the ProviderASSet (Section 3.3 of [I-D.ietf-sidrops-aspa-profile]).

The hash field contains a message digest computed using the aps value (encoded in DER format) as input message.

3.4.4. TrustAnchorState

An instance of TrustAnchorState represents the set of valid Trust Anchor (TA) Certification Authority (CA) resource certificates used by the relying party when producing the CCR.

Each SubjectKeyIdentifier is the 160-bit SHA-1 hash of the value of the DER-encoded ASN.1 bit string of the TA's Subject Public Key, as described in Section 4.8.2 of [RFC6487]. The skis field contains a sequence of Subject Key Identifiers (SKI) sorted in ascending order by interpreting the SKI value as an unsigned 160-bit integer.

The hash field contains a message digest computed using the skis value (encoded in DER format) as input message.

3.4.5. RouterKeyState

An instance of RouterKeyState contains an rksets field which represents the current set of valid BGPsec Router Keys [RFC8205] encoded as a SEQUENCE of RouterKeySet instances. The asID value in each instance of RouterKeySet MUST be unique with respect to other instances of RouterKeySet. Instances of RouterKeySet are sorted by ascending value of asID. Instances of RouterKey are sorted by ascending value of ski by interpreting the SKI value as an unsigned 160-bit integer.

The hash field contains a message digest computed using the rks value (encoded in DER format) as input message.

4. Operational Considerations

Comparing the ManifestState mostRecentUpdate timestamp value with the producedAt timestamp might help offer insight into the timing and propagation delays of the RPKI supply chain.

Given the absence of public keys and fairly repetitive content in RPKI AccessDescription instances, it should be noted CCR content compresses well.

4.1. Verifying CCR file integrity

The integrity of a CCR object can be checked by confirming whether the hash values embedded inside state aspects match the computed hash value of the respective state aspect payload structure.

5. Security Considerations

CCR objects are not signed objects.

6. IANA Considerations

6.1. SMI Security for S/MIME CMS Content Type (1.2.840.113549.1.9.16.1)

IANA has allocated the following in the "SMI Security for S/MIME CMS Content Type (1.2.840.113549.1.9.16.1)" registry:

Decimal	Description	References
54	id-ct-rpkiCanonicalCacheRepresentation	draft-ietf-sidrops-rpki-ccr

Table 1

6.2. RPKI Repository Name Schemes

IANA is requested to add the Canonical Cache Representation file extension to the "RPKI Repository Name Schemes" registry [RFC6481] as follows:

Filename Extension	RPKI Object	Reference
.ccr	Canonical Cache Representation	draft-ietf-sidrops-rpki-ccr

Table 2

6.3. SMI Security for S/MIME Module Identifier (1.2.840.113549.1.9.16.0)

IANA is requested to allocate the following in the "SMI Security for S/MIME Module Identifier (1.2.840.113549.1.9.16.0)" registry:

Decimal	Description	References
TBD	id-mod-rpkiCCR-2025	draft-ietf-sidrops-rpki-ccr

Table 3

6.4. Media Types

IANA is requested to register the media type "application/rpki-ccr" in the "Media Types" registry as follows:

6.4.1. Canonical Cache Representation Media Type

Type name: application
Subtype name: rpki-ccr
Required parameters: N/A
Optional parameters: N/A
Encoding considerations: binary
Security considerations: This media type contains no active content.
Interoperability considerations: N/A
Published specification: draft-ietf-sidrops-rpki-ccr
Applications that use this media type: RPKI operators
Fragment identifier considerations: N/A
Additional information:
Content: This media type is a RPKI Canonical Cache Representation object, as defined in draft-ietf-sidrops-rpki-ccr.
Magic number(s): N/A
File extension(s): .ccr
Macintosh file type code(s): N/A
Person & email address to contact for further information: Job Snijders (job@bsd.nl)
Intended usage: COMMON
Restrictions on usage: N/A
Author: Job Snijders (job@bsd.nl)
Change controller: IETF

7. References

7.1. Normative References

[I-D.ietf-sidrops-aspa-profile]
Azimov, A., Uskov, E., Bush, R., Snijders, J., Housley, R., and B. Maddison, "A Profile for Autonomous System Provider Authorization", Work in Progress, Internet-Draft, draft-ietf-sidrops-aspa-profile-20, 18 August 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-aspa-profile-20>.

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

[RFC6481] Huston, G., Loomans, R., and G. Michaelson, "A Profile for Resource Certificate Repository Structure", RFC 6481, DOI 10.17487/RFC6481, February 2012, <https://www.rfc-editor.org/info/rfc6481>.

[RFC6487] Huston, G., Michaelson, G., and R. Loomans, "A Profile for X.509 PKIX Resource Certificates", RFC 6487, DOI 10.17487/RFC6487, February 2012, <https://www.rfc-editor.org/info/rfc6487>.

[RFC6488] Lepinski, M., Chi, A., and S. Kent, "Signed Object Template for the Resource Public Key Infrastructure (RPKI)", RFC 6488, DOI 10.17487/RFC6488, February 2012, <https://www.rfc-editor.org/info/rfc6488>.

[RFC6811] Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", RFC 6811, DOI 10.17487/RFC6811, January 2013, <https://www.rfc-editor.org/info/rfc6811>.

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

[RFC9286] Austein, R., Huston, G., Kent, S., and M. Lepinski, "Manifests for the Resource Public Key Infrastructure (RPKI)", RFC 9286, DOI 10.17487/RFC9286, June 2022, <https://www.rfc-editor.org/info/rfc9286>.

[RFC9582] Snijders, J., Maddison, B., Lepinski, M., Kong, D., and S. Kent, "A Profile for Route Origin Authorizations (ROAs)", RFC 9582, DOI 10.17487/RFC9582, May 2024, <https://www.rfc-editor.org/info/rfc9582>.

[SHS] National Institute of Standards and Technology, "Secure Hash Standard", March 2012, <https://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf>.

[X.690] ITU-T, "Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, ISO/IEC 8825-1:2021, February 2021, <https://www.itu.int/rec/T-REC-X.690-202102-I/en>.

7.2. Informative References

[I-D.ietf-sidrops-rpki-erik-protocol] Snijders, J., Bruijnzeels, T., Harrison, T., and W. Ohgai, "The Erik Synchronization Protocol for use with the Resource Public Key Infrastructure (RPKI)", Work in Progress, Internet-Draft, draft-ietf-sidrops-rpki-erik-protocol-00, 1 December 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-rpki-erik-protocol-00>.

[RFC5781] Weiler, S., Ward, D., and R. Housley, "The rsync URI Scheme", RFC 5781, DOI 10.17487/RFC5781, February 2010, <https://www.rfc-editor.org/info/rfc5781>.

[RFC8182] Bruijnzeels, T., Muravskiy, O., Weber, B., and R. Austein, "The RPKI Repository Delta Protocol (RRDP)", RFC 8182, DOI 10.17487/RFC8182, July 2017, <https://www.rfc-editor.org/info/rfc8182>.

[RFC8205] Lepinski, M., Ed. and K. Sriram, Ed., "BGPsec Protocol Specification", RFC 8205, DOI 10.17487/RFC8205, September 2017, <https://www.rfc-editor.org/info/rfc8205>.

[rpki-client] Jeker, C., Dzonsons, K., Buehler, T., and J. Snijders, "rpki-client", December 2025, <https://www.rpki-client.org/>.

[rpkitouch] Snijders, J., "rpki-client", December 2025, <https://www.github.com/job/rpkitouch>.

Appendix A. Acknowledgements

The authors wish to thank Russ Housley and Luuk Hendriks for their generous feedback on this specification.

Appendix B. Example CCR

The below is a Base64-encoded example CCR object. For a more elaborate example based on the global RPKI, see the URL in Appendix C.

MIIN9wYLKoZIhvcNAQkQATaggg3mMIIN4jALBglghkgBZQMEAgEYDzIwMjUxMjA0MTAzO
TIyWqGCCd4wggnaMIIJozCB0QQgAn4v94Lj6dIrJVXA6nPyEXUf2KSwui6SPTq5B4TuRu
ACAgfOBBSQIY6AGlMllem3HGQ2hOoF+Wv18wICBMAYDzIwMjUxMjA0MDIwMTA5WjB+MHw
GCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxU
LzNmLzFiNjYyNC04NDQxLTRkMDEtOTZlMy02MDE4MTJlZjQyOGIvMS9rQ0dPZ0JwVEpaW
HB0eHhrTm9UcUJmbHI5Zk0ubWZ0MIHRBCACgrfBbvv/vMbbn2Ix5BHOXUqO+1b3/eDjEx
kW+c8crgICCBgEFAUrhklp6WgHZImQFeFj7Od7tMt2AgILSBgPMjAyNTEyMDQxMDAwMDl
aMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RF
RkFVTFQvMjQvNDc1OWY5LTdmZGYtNDkxNi1hOTk1LTY5NzBlY2Q5YTYxMC8xL0JTdUdTV
25wYUFka2laQVY0V1BzNTN1MHkzWS5tZnQwgdEEIAKDa5Xc2Ckfla7w7za0h40htY2Gv9
1o0fT/rzNm39EBAgIHzgQUZZq64rDK6GxBlrAgdluCOiAyB/wCAhdcGA8yMDI1MTIwNDA
3MDAxMFowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRv
cnkvREVGQVVMVC85Ni82OTRkOWItMGUxYy00M2FkLWFjOTgtMDJhYWM4YjU5NmRjLzEvW
lpxNjRyREs2R3hCbHJBZ2RsdUNPaUF5Ql93Lm1mdDCCATsEIAKJwo+XaFAxvIQbXPID/4
mkW2UQmjHTsQcG0KokS9BKAgIJOAQU185GL3ELzfDYdT2D2OojR5RqfGECFAENDJ9DKFh
IBelh+Yl+p9QFY/HcGA8yMDI1MTIwMzIyMDAwOFowgdUwgdIGCCsGAQUFBzALhoHFcnN5
bmM6Ly9ycGtpLmFyaW4ubmV0L3JlcG9zaXRvcnkvYXJpbi1ycGtpLXRhLzVlNGEyM2VhL
WU4MGEtNDAzZS1iMDhjLTIxNzFkYTIxNTdkMy9kOWQxNTcyZi02Y2JiLTRjZjctYjU5OS
1lOWQwZTk4MWQ5YmYvOTk4Y2U2YzYtZGU3Ni00MDI4LWE2ZmUtZWEwM2NmZmZmYmNiLzk
5OGNlNmM2LWRlNzYtNDAyOC1hNmZlLWVhMDNjZmZmZmJjYi5tZnQwgdEEIAKLk3NcLwUp
ybu+/VTVy3jDiw1XMRgMfuUg1L+DkSkrAgIIGAQUUH5YKtyTadqK6F3ZNXQBIwgcfu0CA
goWGA8yMDI1MTIwNDA0MDA0M1owfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcG
UubmV0L3JlcG9zaXRvcnkvREVGQVVMVC83NS82ZWQ0MzQtY2Q1MS00MTUyLWFhNDMtMDU
2YmFlMjcyODhlLzEvVUg1WUt0eVRhZHFLNkYzWk5YUUJJd2djZnUwLm1mdDCCATsEIAKM
2e5JA/52WWa71nAV9G+ChOWEv0nksETnQFPhIwvLAgIJiwQUGggBUYL46DBD52qzRc1Uc
qS0zfwCFAENDJ9DKFhAc6gwUfTYJdAsiOPbGA8yMDI1MTIwMzIzMDAwM1owgdUwgdIGCC
sGAQUFBzALhoHFcnN5bmM6Ly9ycGtpLmFyaW4ubmV0L3JlcG9zaXRvcnkvYXJpbi1ycGt
pLXRhLzVlNGEyM2VhLWU4MGEtNDAzZS1iMDhjLTIxNzFkYTIxNTdkMy81MjFlYjMzZi05
NjcyLTRjZDktYWNjZS0xMzcyMjdlOTcxYWMvNGU4N2Q4M2YtYjUwYy00YTY1LTlkOWEtY
zI4NDU4ZmMxOTc5LzRlODdkODNmLWI1MGMtNGE2NS05ZDlhLWMyODQ1OGZjMTk3OS5tZn
QwggFTBCACkKcTyzxq9pGovZfaazRbYvlJhP5FrMqFdnWHLxv37QICCUcEFHa0G4uxYRm
zlpyY5QtHr7bF/4JzAhQBDQyfQyhYQHOoNVDDogCyPg+0rRgPMjAyNTEyMDQwMTAwMDNa
MIHVMIHSBggrBgEFBQcwC4aBxXJzeW5jOi8vcnBraS5hcmluLm5ldC9yZXBvc2l0b3J5L
2FyaW4tcnBraS10YS81ZTRhMjNlYS1lODBhLTQwM2UtYjA4Yy0yMTcxZGEyMTU3ZDMvNz
ZmZTExZDQtZDM1Mi00OTk0LThmNmMtZDZjOTFiMGI4NDE1LzcyMjViNDE1LThhZTAtNDU
yMy1iMzdmLTc0ZWQ3ODA2NzZhYS83MjI1YjQxNS04YWUwLTQ1MjMtYjM3Zi03NGVkNzgw
Njc2YWEubWZ0MBYEFBjAkk0jHaMBlRYLJe7mMn60Awb4MIIBOwQgApKZUyrzgxwgooY27

```
   kVNyU/GcfKMveLqnEy9eC1eSwwCAgoqBBSNdoMwkedLBIRD98SPJFBQ0ZiW9AIUAQ0Mn0
   MoWEfSt9vLHRt0f4E9Bx0YDzIwMjUxMjA0MDIwMjA5WjCB1TCB0gYIKwYBBQUHMAuGgcV
   yc3luYzovL3Jwa2kuYXJpbi5uZXQvcmVwb3NpdG9yeS9hcmluLXJwa2ktdGEvNWU0YTIz
   ZWEtZTgwYS00MDNlLWIwOGMtMjE3MWRhMjE1N2QzLzY5ZmQwMTU2LWJiMWYtNDhiNi1iZ
   jMyLWM5NDkyMjg2ZjE5NS80YzNiYjdlNC1jOTdlLTQzYWItYTZkZC1jYmYwZTY0MzRjNW
   IvNGMzYmI3ZTQtYzk3ZS00M2FiLWE2ZGQtY2JmMGU2NDM0YzViLm1mdDCCATsEIAKTgAc
   MIFK5KQy6hB+48lt24P5LOju8QSUJXNjqOozwAgIJOAQUuR/cF6KhF1ZW51AnlhafP7ra
   GygCFAENDJ9DKFhEouhGggdtM1PWJqI7GA8yMDI1MTIwNDA5MDAwOFowgdUwgdIGCCsGA
   QUFBzALhoHFcnN5bmM6Ly9ycGtpLmFyaW4ubmV0L3JlcG9zaXRvcnkvYXJpbi1ycGtpLX
   RhLzVlNGEyM2VhLWU4MGEtNDAzZS1iMDhjLTIxNzFkYTIxNTdkMy80YWI3YWU0ZC1iZDd
   iLTRiMzMtOWE4OC01YjIyZDJhODMzN2QvNmY0ZDhlNmItZTY2ZS00NmY5LThlZjItZjJk
   YTE5ODFjYmNhLzZmNGQ4ZTZiLWU2NmUtNDZmOS04ZWYyLWYyZGExOTgxY2JjYS5tZnQYD
   zIwMjUxMjA0MTAwMDA5WgQgaNOQqYiZBV7B7dtdF6T9PhQFyhn6h97ab7mkUePReaaigg
   HeMIIB2jCCAbQwZAIBBzBfMEgEAgABMEIwCQMEAMAjXgIBIDAJAwQAwEMrAgEgMAkDBAD
   CIEUCASAwCQMEAcIg2gIBIDAJAwQAwiKKAgEgMAkDBAHCPVwCASAwEwQCAAIwDTALAwUD
   Kgs7QAICAIAwgZsCAiBbMIGUMHYEAgABMHAwBgMEAFvQIjAGAwQAXo7wMAYDBANejvAwB
   gMEAF6O8TAGAwQAXo7yMAYDBABejvQwBgMEAF6O9TAGAwQAXo72MAYDBABejvcwBgMEAL
   k04DAGAwQCuTTgMAYDBAC5NOEwBgMEALk04jAGAwQAuTTjMBoEAgACMBQwCQMHACABBng
   GiDAHAwUAKgIImDCBrQICPMowgaYwVwQCAAEwUTAGAwQAQ931MAYDBACl/uEwCQMEAKX+
   /wIBIDAGAwQAwJOoMAkDBAHGOgICARgwCQMEAcwCHgIBGDAGAwQA0RgBMAYDBADRGAUwB
   gMEANEYCTBLBAIAAjBFMAwDBwEgAQQYFE4CAUAwCQMHACABBnwgjDAJAwcAIAEHKBgIMA
   kDBwAmB/rgAkUwCQMHACoOskAAADAJAwcAKg6yQAEYBCDQKq45jwi7kIlRM6oQqIdw8Ck
   6H0Wn23dFazmtj/S28KOBjDCBiTBlMAoCAghJMAQCAg0FMAkCAhGMMAMCAQAwDwICEfkw
   CQICIGoCAwDjAzAjAgIZGDAdAgIArgICBPkCAgUTAgIZPQICGmoCAhquAgMCJ4kwFgICG
   ncwEAICAK4CAhg8AgIbGwICMuYEICz1Hxj/8Ur8yZsJDt5IGPn/pGKgaURkFZUkoheP7O
   iDpFIwUDAsBBQT1PJPmp/NmNs2+TBjGAjIjzl0vAQU6FUrH9bRpPfkBMbY5WgNHrwWP8M
   EIKHmyNKlH4f3f7a1i6qTkZmQEBEAqGEA/uH4coZH5qAMpYIBGTCCARUwgfAwge0CAjzK
   MIHmMHEEFF1CUOLYHURI2KKe/OkdKf8HXsniMFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQ
   gAEgFcjQ/g//LAQerAH2Mpp+GucoDAGBbhIqD33wNPsXxnAGb+mtZ7XQrVO9DQ6UlASht
   ig5+QfEKpTtFgiqfiAFTBxBBS+iJtV0Lc3OX11xJ9IW4WPqYrRHzBZMBMGByqGSM49AgE
   GCCqGSM49AwEHA0IABOBcSa9J9m7sdbl9RL5fkFsGWLyGnT4y7hV9psairgBlISp6+1Sy
   w4KxPvpfaeXh9pFkzVQDdthVFN3W/0SqRNsEILpftEnO+2ugDzYSeWKi7qboZ/6FErvdr
   enG5Li8FsHS
```

It decodes as follows:

```
   =============== NOTE: '\' line wrapping per RFC 8792 ================

   File:                    example.ccr
   Hash identifier:         wHMUl0+oVEBXXPPxp+0XUhaHaNb2phSO0dScm+emGx8=
   CCR produced at:         Thu 04 Dec 2025 10:39:22 +0000
   Manifest state hash:     NjhEMzkwQTk4ODk5MDU1RUMxRUREQjVEMTdBNEZEM0U=
   Manifest last update:    Thu 04 Dec 2025 10:00:09 +0000
   Manifest instances:
                            hash:An4v94Lj6dIrJVXA6nPyEXUf2KSwui6SPTq5B4T\
   uRuA= size:1998 aki:90218E801A532595E9B71C643684EA05F96BF5F3 seqnum:\
   04C0 thisupdate:1764813669 sia:rsync://rpki.ripe.net/repository/DEFA\
   ULT/3f/1b6624-8441-4d01-96e3-601812ef428b/1/kCGOgBpTJZXptxxkNoTqBflr\
```

```
   9fM.mft
                            hash:AoK3wW77/7zG259iMeQRzl1KjvtW9/3g4xMZFvn\
   PHK4= size:2072 aki:052B864969E9680764899015E163ECE77BB4CB76 seqnum:\
   0B48 thisupdate:1764842409 sia:rsync://rpki.ripe.net/repository/DEFA\
   ULT/24/4759f9-7fdf-4916-a995-6970ecd9a610/1/BSuGSWnpaAdkiZAV4WPs53u0\
   y3Y.mft
                            hash:AoNrldzYKR+VrvDvNrSHjSG1jYa/3WjR9P+vM2b\
   f0QE= size:1998 aki:659ABAE2B0CAE86C4196B020765B823A203207FC seqnum:\
   175C thisupdate:1764831610 sia:rsync://rpki.ripe.net/repository/DEFA\
   ULT/96/694d9b-0e1c-43ad-ac98-02aac8b596dc/1/ZZq64rDK6GxBlrAgdluCOiAy\
   B_w.mft
                            hash:AonCj5doUDG8hBtc8gP/iaRbZRCaMdOxBwbQqiR\
   L0Eo= size:2360 aki:D7CE462F710BCDF0D8753D83D8EA2347946A7C61 seqnum:\
   010D0C9F4328584805E961F9897EA7D40563F1DC thisupdate:1764799208 sia:r\
   sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
   -2171da2157d3/d9d1572f-6cbb-4cf7-b599-e9d0e981d9bf/998ce6c6-de76-402\
   8-a6fe-ea03cffffbcb/998ce6c6-de76-4028-a6fe-ea03cffffbcb.mft
                            hash:AouTc1wvBSnJu779VNXLeMOLDVcxGAx+5SDUv4O\
   RKSs= size:2072 aki:507E582ADC9369DA8AE85DD935740123081C7EED seqnum:\
   0A16 thisupdate:1764820843 sia:rsync://rpki.ripe.net/repository/DEFA\
   ULT/75/6ed434-cd51-4152-aa43-056bae27288e/1/UH5YKtyTadqK6F3ZNXQBIwgc\
   fu0.mft
                            hash:AozZ7kkD/nZZZrvWcBX0b4KE5YS/SeSwROdAU+E\
   jC8s= size:2443 aki:1A08015182F8E83043E76AB345CD5472A4B4CDFC seqnum:\
   010D0C9F4328584073A83051F4D825D02C88E3DB thisupdate:1764802803 sia:r\
   sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
   -2171da2157d3/521eb33f-9672-4cd9-acce-137227e971ac/4e87d83f-b50c-4a6\
   5-9d9a-c28458fc1979/4e87d83f-b50c-4a65-9d9a-c28458fc1979.mft
                            hash:ApCnE8s8avaRqL2X2ms0W2L5SYT+RazKhXZ1hy8\
   b9+0= size:2375 aki:76B41B8BB16119B3969C98E50B47AFB6C5FF8273 seqnum:\
   010D0C9F4328584073A83550C3A200B23E0FB4AD thisupdate:1764810003 sia:r\
   sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
   -2171da2157d3/76fe11d4-d352-4994-8f6c-d6c91b0b8415/7225b415-8ae0-452\
   3-b37f-74ed780676aa/7225b415-8ae0-4523-b37f-74ed780676aa.mft subordi\
   nates:18C0924D231DA30195160B25EEE6327EB40306F8
                            hash:ApKZUyrzgxwgooY27kVNyU/GcfKMveLqnEy9eC1\
   eSww= size:2602 aki:8D76833091E74B048443F7C48F245050D19896F4 seqnum:\
   010D0C9F43285847D2B7DBCB1D1B747F813D071D thisupdate:1764813729 sia:r\
   sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
   -2171da2157d3/69fd0156-bb1f-48b6-bf32-c9492286f195/4c3bb7e4-c97e-43a\
   b-a6dd-cbf0e6434c5b/4c3bb7e4-c97e-43ab-a6dd-cbf0e6434c5b.mft
                            hash:ApOABwwgUrkpDLqEH7jyW3bg/ks6O7xBJQlc2Oo\
   6jPA= size:2360 aki:B91FDC17A2A1175656E7502796169F3FBADA1B28 seqnum:\
   010D0C9F43285844A2E84682076D3353D626A23B thisupdate:1764838808 sia:r\
   sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
   -2171da2157d3/4ab7ae4d-bd7b-4b33-9a88-5b22d2a8337d/6f4d8e6b-e66e-46f\
   9-8ef2-f2da1981cbca/6f4d8e6b-e66e-46f9-8ef2-f2da1981cbca.mft
   ROA payload state hash: RDAyQUFFMzk4RjA4QkI5MDg5NTEzM0FBMTBBODg3NzA=
```

```
   ROA payload entries:
                           192.35.94.0/24-32 AS 7
                           192.67.43.0/24-32 AS 7
                           194.32.69.0/24-32 AS 7
                           194.32.218.0/23-32 AS 7
                           194.34.138.0/24-32 AS 7
                           194.61.92.0/23-32 AS 7
                           2a0b:3b40::/29-128 AS 7
                           91.208.34.0/24 AS 8283
                           94.142.240.0/24 AS 8283
                           94.142.240.0/21 AS 8283
                           94.142.241.0/24 AS 8283
                           94.142.242.0/24 AS 8283
                           94.142.244.0/24 AS 8283
                           94.142.245.0/24 AS 8283
                           94.142.246.0/24 AS 8283
                           94.142.247.0/24 AS 8283
                           185.52.224.0/24 AS 8283
                           185.52.224.0/22 AS 8283
                           185.52.225.0/24 AS 8283
                           185.52.226.0/24 AS 8283
                           185.52.227.0/24 AS 8283
                           2001:678:688::/48 AS 8283
                           2a02:898::/32 AS 8283
                           67.221.245.0/24 AS 15562
                           165.254.225.0/24 AS 15562
                           165.254.255.0/24-32 AS 15562
                           192.147.168.0/24 AS 15562
                           198.58.2.0/23-24 AS 15562
                           204.2.30.0/23-24 AS 15562
                           209.24.1.0/24 AS 15562
                           209.24.5.0/24 AS 15562
                           209.24.9.0/24 AS 15562
                           2001:418:144e::/47-64 AS 15562
                           2001:67c:208c::/48 AS 15562
                           2001:728:1808::/48 AS 15562
                           2607:fae0:245::/48 AS 15562
                           2a0e:b240::/48 AS 15562
                           2a0e:b240:118::/48 AS 15562
   ASPA payload state hash:MkNGNTFGMThGRkYxNEFGQ0M5OUIwOTBFREU0ODE4Rjk=
   ASPA payload entries:
                           customer: 2121 providers: 3333
                           customer: 4492 providers: 0
                           customer: 4601 providers: 8298, 58115
                           customer: 6424 providers: 174, 1273, 1299, 6\
   461, 6762, 6830, 141193
                           customer: 6775 providers: 174, 6204, 6939, 1\
   3030
```

```
   Trust anchor state hash:QTFFNkM4RDJBNTFGODdGNzdGQjZCNThCQUE5MzkxOTk=
   Trust anchor keyids:    13D4F24F9A9FCD98DB36F930631808C88F3974BC, E8\
   552B1FD6D1A4F7E404C6D8E5680D1EBC163FC3
   Router key state hash:  QkE1RkI0NDlDRUZCNkJBMDBGMzYxMjc5NjJBMkVFQTY=
   Router keys:
                           asid:15562 ski:5D4250E2D81D4448D8A29EFCE91D2\
   9FF075EC9E2 pubkey:MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAEgFcjQ/g//LAQe\
   rAH2Mpp+GucoDAGBbhIqD33wNPsXxnAGb+mtZ7XQrVO9DQ6UlAShtig5+QfEKpTtFgiq\
   fiAFQ==
                           asid:15562 ski:BE889B55D0B737397D75C49F485B8\
   58FA98AD11F pubkey:MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAE4FxJr0n2bux1u\
   X1Evl+QWwZYvIadPjLuFX2mxqKuAGUhKnr7VLLDgrE++l9p5eH2kWTNVAN22FUU3db/R\
   KpE2w==
   Validation:             N/A
```

Appendix C. Implementation status

This section is to be removed before publishing as an RFC.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in RFC 7942. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to RFC 7942, "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

* Example .ccr files were created by Job Snijders. A current example CCR (regenerated every few minutes) is available here: https://console.rpki-client.org/rpki.ccr

* A CCR serializer and deserializer implementation based on [rpki-client] was provided by Job Snijders and Theo Buehler.

* Another CCR serializer and deserializer implementation based on [rpkitouch] was provided by Job Snijders.

Authors' Addresses

Job Snijders
BSD Software Development
Amsterdam
Netherlands
Email: job@bsd.nl
URI: https://www.bsd.nl

Bart Bakker
RIPE NCC
Netherlands
Email: bbakker@ripe.net

Tim Bruijnzeels
RIPE NCC
Netherlands
Email: tbruijnzeels@ripe.net

Theo Buehler
OpenBSD
Switzerland
Email: tb@openbsd.org