SIDROPS — J. Snijders, BSD
Internet-Draft — B. Bakker
Intended status: Standards Track — T. Bruijnzeels, RIPE NCC
Expires: 7 November 2026 — T. Buehler, OpenBSD
6 May 2026

A Profile for Resource Public Key Infrastructure (RPKI) Canonical Cache Representation (CCR)

draft-ietf-sidrops-rpki-ccr-04

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 RPKI cache at a particular point in time. The CCR profile is a compact and versatile format well-suited for applications such as audit trails, analytics pipelines, and validated payload dissemination.

Table of Contents

1. Introduction

Resource Public Key Infrastructure (RPKI) operators often wish to analyze Certification Authority (CA) and Relying Party (RP) behavior by inspecting validation outcomes. To this end, Canonical Cache Representation (CCR) was developed to capture and archive RPKI validation states in a standardized data representation.

CCR offers a compact and versatile format well-suited for applications such as audit trails, analytics pipelines, and validated payload dissemination. A validated cache contains all RPKI objects that the 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 in a reproducible manner.

This document formally specifies the CCR content type for use with the RPKI and provides test vectors.

1.1. History

The format was initially designed to support comparative analysis of multiple RP instances using a variety of RPKI transport protocols ([RFC5781], [RFC8182], and [I-D.ietf-sidrops-rpki-erik-protocol]).

1.2. 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 field contains 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) }

  CAS, PAS
  FROM RPKI-ASPA-2023 -- in [draft-ietf-sidrops-aspa-profile]
    { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
     pkcs-9(9) smime(16) modules(0) id-mod-rpki-aspa-2023(TBD) }

  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      CAS,
     providers         SEQUENCE (SIZE(1..MAX)) OF PAS }

   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 and the parameters field MUST be absent (Section 2 of [RFC5754]).

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, and 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").

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

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. Use Cases

This section describes a number of applications for the CCR format across different contexts.

4.1. Constructing Consistent Views on Distributed Data

This section describes a use case for CCRs in the context of distributed systems.

Assuming CAs issue Manifests in accordance with Section 5 of [RFC9286], a ManifestInstance can be considered a state-based Conflict-free Replicated Data Type ([CRDT]), meaning that ManifestInstance sets contain sufficient information to form a monotonic semilattice.

The implication is that ManifestState instances from multiple CCRs produced by multiple different RPs at different times can safely be merged in order to construct an internally consistent view of the RPKI distributed database.

The reconciled merge result can be useful, for example, as a backend for Erik Synchronization relays ([I-D.ietf-sidrops-rpki-erik-protocol]) which execute separate validation processes for different Trust Anchors and varying maximum certificate chain depths.

4.2. Data Collection

Operators have an interest in determining how the global RPKI is viewed from the perspectives of several different locations around the Internet. As CCR allows for point-in-time capture and later reconstruction and analysis, it found use in multi-perspective collector methods such as described RPKISPOOL [I-D.snijders-rpkispool-format].

An example of a large-scale CCR-based RPKI data archival project is [RPKIViews].

5. Operational Considerations

This section covers operational considerations.

5.1. CCR file integrity

The integrity of a CCR file can be checked by confirming whether the hash value embedded inside each state aspect matches the computed hash value of the respective state aspect payload structure. Readers MUST verify the integrity of CCR files and stop further processing on failure.

5.2. Efficiency

CCRs compress very well due to its data layout characteristics: the content contains repetitive sequences, does not contain high entropy data such as public keys, and is consistently ordered. Readers and writers of CCR data are RECOMMENDED to support data compression using Gzip ([RFC1952]) in context of durable storage.

6. Security Considerations

The CCR format utilizes a structure that can store information about the state of a given RPKI cache at a particular moment in time. The fields defined in this specification are of a descriptive nature and provide information that is useful to facilitate the analysis of RPKI data. As such, these fields do not in themselves create additional security risks, since the fields are not used to induce any particular behavior by the recipient application.

Readers MUST check contextual bounds on all fields appropriately and stop further processing on failure. E.g., the maxLength element in a ROAIPAddress cannot contain an integer smaller than the length of the accompanying prefix, the manifestNumber field is cannot be longer than 20 octets, etc.

The CCR format contains no executable code, and it does not define any extensible areas that could be used to store such code.

CCRs are not signed objects. RPKI information is normally public and does not call for confidentiality protection. Ascertaining the provenance (and thus authenticity) of any given CCR is out-of-scope for this document.

7. IANA Considerations

7.1. 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

7.2. 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

7.3. 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

7.4. Media Types

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

7.4.1. 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

Type name: application
Subtype name: rpki-ccr+gz
Content: This media type is a Gzip compressed RPKI Canonical Cache Representation object, as defined in draft-ietf-sidrops-rpki-ccr.
Magic number(s): N/A
File extension(s): .ccr.gz
References: RFC1952, RFC6713
Encoding considerations: gzip is a binary encoding

8. References

8.1. Normative References

[I-D.ietf-sidrops-aspa-profile]
Snijders, J., Azimov, A., Uskov, E., Bush, R., Housley, R., and B. Maddison, "A Profile for Autonomous System Provider Authorization", Work in Progress, Internet-Draft, draft-ietf-sidrops-aspa-profile-26, 19 April 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-aspa-profile-26>.

[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>.

[RFC5754] Turner, S., "Using SHA2 Algorithms with Cryptographic Message Syntax", RFC 5754, DOI 10.17487/RFC5754, January 2010, <https://www.rfc-editor.org/info/rfc5754>.

[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>.

8.2. Informative References

[CRDT] Shapiro, M., Pregui巽a, N., Baquero, C., and M. Zawirski, "Conflict-free Replicated Data Types", INRIA RR-7687, July 2011, <https://inria.hal.science/inria-00609399>.

[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-04, 17 March 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-rpki-erik-protocol-04>.

[I-D.snijders-rpkispool-format]
Snijders, J. and F. Vompe, "The RPKISPOOL Format for Materializing Resource Public Key Infrastructure (RPKI) Data", Work in Progress, Internet-Draft, draft-snijders-rpkispool-format-00, 2 March 2026, <https://datatracker.ietf.org/doc/html/draft-snijders-rpkispool-format-00>.

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

[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/>.

[rpki-commons]
NCC, R., "rpki-commons", April 2026, <https://github.com/RIPE-NCC/rpki-commons>.

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

[RPKIViews]
Snijders, J., "The RPKIViews Project", April 2026, <https://www.rpkiviews.org/>.

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.

```
MIIP/wYLKoZIhvcNAQkQATaggg/uMIIP6jALBglghkgBZQMEAgEYDzIwMjYwNDExMDgwN
DMxWqGCC9kwggvVMIILnjCB0QQgAAA2wRwPsxllQz3CGSuUSNg95LD7ve8TkQG8oJfZf/
QCAgfOBBRGOHxWszH/hLwQ2KyQ4eLBbxcjRQICGLIYDzIwMjYwNDEwMjMwMTUxWjB+MHw
GCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxU
LzQ4LzFiNDBmZi1iMWUxLTQ5NTEtOTE2NS0yM2JiMzlhODM0ODEvMS9Samg4VnJNeF80U
zhFTmlza09IaXdXOFhJMFUubWZ0MIIBOwQgAAFxGHgJjLarAoLN6aV4ByTazpqHNrQ4xD
jc5eXRQrYCAgk4BBTA1zPgXUwFbjp+lDMtxGvoAUhoigIUAQ0Mn0MoWEPsKztq6RnIjIf
zkgAYDzIwMjYwNDEwMjIwMDAzWjCB1TCB0gYIKwYBBQUHMAuGgcVyc3luYzovL3Jwa2ku
YXJpbi5uZXQvcmVwb3NpdG9yeS9hcmluLXJwa2ktdGEvNWU0YTIzZWEtZTgwYS00MDNlL
WIwOGMtMjE3MWRhMjE1N2QzLzg3MWRhNDBmLTc5M2EtNGE0NS1hMGE5LTk3ODE0ODMyMW
EwNy9hMTIwYjVkNC1kYTU2LTQ5YjEtOGFlMS03OTg3YTZhZmRlOTkvYTEyMGI1ZDQtZGE
1Ni00OWIxLThhZTEtNzk4N2E2YWZkZTk5Lm1mdDCCATsEIAAFujQiTYR4XP+wPQa3rvHa
Z9sVxQu9TikQPbvcSrGUAgIJOAQUrhIoBf2wm/Wx39wUmFrEDwf0PoUCFAENDJ9DKFhLn
OZ6Otp9FmCbky+PGA8yMDI2MDQxMTAzMDAwM1owgdUwgdIGCCsGAQUFBzALhoHFcnN5bm
M6Ly9ycGtpLmFyaW4ubmV0L3JlcG9zaXRvcnkvYXJpbi1ycGtpLXRhLzVlNGEyM2VhLWU
4MGEtNDAzZS1iMDhjLTIxNzFkYTIxNTdkMy84NWUwNmVhZi0zN2E3LTQ1ODgtYWJlYS1l
NDkwOWVmMWI0ZTIvYjJjNThiYTktMzM5My00ZjRjLWFiZTEtYjIzMWEzZWEyNzAwL2IyY
zU4YmE5LTMzOTMtNGY0Yy1hYmUxLWIyMzFhM2VhMjcwMC5tZnQwgdEEIAAGt84u5ZglXr
x63YwuEZbT7Vu8lb+MPP+inyrftREGAgIHzgQUFrGYtu469o3rwjR6Xpj3Efsw8osCAgW
2GA8yMDI2MDQxMTAzMDIxM1owfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUu
bmV0L3JlcG9zaXRvcnkvREVGQVVMVC8zYS8yMmFlMTQtZTQ1ZS00ZWVlLWJkMDctNDQ4M
mFkYTIzMmUzLzEvRnJHWXR1NDY5bzNyd2pSNlhwajNFZnN3OG9zLm1mdDCCATsEIAAHY6
f9bYuKrhs+7nHfxGf4NvKaoK3bCDdhaB7CPzv7AgIJiwQUAnJUlPp8gUJ+xZ8HE/O5sGj
ryMICFAENDJ9DKFhCdi5c30h0SNvj1lMGGA8yMDI2MDQxMDIzMDAwM1owgdUwgdIGCCsG
AQUFBzALhoHFcnN5bmM6Ly9ycGtpLmFyaW4ubmV0L3JlcG9zaXRvcnkvYXJpbi1ycGtpL
XRhLzVlNGEyM2VhLWU4MGEtNDAzZS1iMDhjLTIxNzFkYTIxNTdkMy81YjdmYjEyMi1kZm
RmLTRjMGMtYjkwZC0zYmM3YTVmZWI4MmIvZmRjMzM2NWEtMThlYS00Njk2LWFkOGMtNmI
2NmEzZTE1MmI3L2ZkYzMzNjVhLTE4ZWEtNDY5Ni1hZDhjLTZiNjZhM2UxNTJiNy5tZnQw
gdEEIAAIAYHbABeU9ubNQzM6b+nZlZJPxfMnuoFMasNkPlFrAgIHhAQUONY8X84e8J5L8
```

s/JS7JQn9X/UJwCAgbkGA8yMDI2MDQxMTA2MDExOVowfjB8BggrBgEFBQcwC4ZwcnN5bm
M6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC9iZi9lZTNkNzMtOTcyOS0
0ZGE0LThiYzctNjdjNDQyZDZhODUwLzEvT05ZOFg4NGU4SjVMOHNfSlM3SlFuOVhfVUp3
Lm1mdDCB0QQgAAgGTG8qcTnrJ+s3v2qwsYhicint7+AAzZvEBKS0/CgCAggYBBRz4Veyk
Yytyopan7xm6Xdgim314QICGLsYDzIwMjYwNDExMDQwMDIwWjB+MHwGCCsGAQUFBzALhn
Byc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxULzc5L2JiY2Q1My1
jNGY4LTQyNDUtYmI5MC0wMGExNTRiOGVjYjEvMS9jLUZYc3BHTXJjcUtXcC04WnVsM1lJ
cHQ5ZUUubWZ0MIIBOwQgAAxlL+UgVK9Dx4hdFmtC1Io4cnNXa+WSktfxHNDM2dQCAgk4B
BSebvQFHAvG2whHV6ijen1ZKa7QMwIUAQ0Mn0MoWEfYv17xDpMYoOXHJXMYDzIwMjYwND
EwMTgwMDAzWjCB1TCB0gYIKwYBBQUHMAuGgcVyc3luYzovL3Jwa2kuYXJpbi5uZXQvcmV
wb3NpdG9yeS9hcmluLXJwa2ktdGEvNWU0YTIzZWEtZTgwYS00MDNlLWIwOGMtMjE3MWRh
MjE1N2QzLzY5ZmQwMTU2LWJiMWYtNDhiNi1iZjMyLWM5NDkyMjg2ZjE5NS9hZmIzNTExY
i1mMWI2LTQ0YWQtOWMzNi1mNjU3NmE1NDAwZGYvYWZiMzUxMWItZjFiNi00NGFkLTljMz
YtZjY1NzZhNTQwMGRmLm1mdDCB0QQgABHMuie2PQZfMvLKOxKMHfjD1eaPqk9g7ApjxVq
RekoCAgfOBBTad/oQDWzSiOVEx8tMfU0Yh5B57QICA0QYDzIwMjYwNDExMDAwMTAwWjB+
MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBV
UxUL2RjL2UzMWFkNS1lZjc0LTQwYTQtOWEwZC1kZjg3MmZiMjY5Y2YvMS8ybmY2RUExcz
BvamxSTWZMVEgxTkdJZVFlZTAubWZ0MIIBOwQgABRkjCfKzkq+VljmLDPOTrRzb9XCSHS
4+MdJ85h48CkCAgk4BBROQoyVjC53vymSTEMHykx6r8ysjwIUAQ0Mn0MoWEV8jtR6N+Vu
HCt5JiEYDzIwMjYwNDExMDgwMDAzWjCB1TCB0gYIKwYBBQUHMAuGgcVyc3luYzovL3Jwa
2kuYXJpbi5uZXQvcmVwb3NpdG9yeS9hcmluLXJwa2ktdGEvNWU0YTIzZWEtZTgwYS00MD
NlLWIwOGMtMjE3MWRhMjE1N2QzLzRhYjdhZTRkLWJkN2ItNGIzMy05YTg4LTViMjJkMmE
4MzM3ZC8wMjJkMDI2OS03ZDU2LTQ1Y2MtODE2Ny1iMWU3YWIxM2YxZjQvMDIyZDAyNjkt
N2Q1Ni00NWNjLTgxNjctYjFlN2FiMTNmMWY0Lm1mdDCCATsEIAAZpDK5Kgx6uUVKrKMks
brMn/uM9zx0ScgmSrNPAeccAgII6AQU1rp+M1W1y/Z0A5I2TKkhN5JBwCcCFAENDJ9DKF
hBD/Z3xJ0CPoX7w2cfGA8yMDI2MDQxMDE1MDAwM1owgdUwgdIGCCsGAQUFBzALhoHFcnN
5bmM6Ly9ycGtpLmFyaW4ubmV0L3JlcG9zaXRvcnkvYXJpbi1ycGtpLXRhLzVlNGEyM2Vh
LWU4MGEtNDAzZS1iMDhjLTIxNzFkYTIxNTdkMy8yYTI0Njk0Ny0yZDYyLTRhNmMtYmEwN
S04NzE4N2YwMDk5YjIvMGE5NGI0NjAtNzQ0MS00OTVlLTkzNTgtNDc1ZWE3OTVlY2M2Lz
BhOTRiNDYwLTc0NDEtNDk1ZS05MzU4LTQ3NWVhNzk1ZWNjNi5tZnQYDzIwMjYwNDExMDg
wMDAzWgQg8bXskzbWa1oCoQYF1VnbQskxegvOO2eyS67YnkY29wiggHeMIIB2jCCAbQw
ZAIBBzBfMEgEAgABMEIwCQMEAMAjXgIBIDAJAwQAwEMrAgEgMAkDBADCIEUCASAwCQMEA
cIg2gIBIDAJAwQAwiKKAgEgMAkDBAHCPVwCASAwEwQCAAIwDTALAwUDKgs7QAICAIAwgZ
sCAiBbMIGUMHYEAgABMHAwBgMEAFvQIjAGAwQDXo7wMAYDBABejvAwBgMEAF6O8TAGAwQ
AXo7yMAYDBABejvQwBgMEAF6O9TAGAwQAXo72MAYDBABejvcwBgMEArk04DAGAwQAuTTg
MAYDBAC5NOEwBgMEALk04jAGAwQAuTTjMBoEAgACMBQwCQMHACABBngGiDAHAwUAKgIIm
DCBrQICPMowgaYwVwQCAAEwUTAGAwQAQ931MAYDBACl/uEwCQMEAKX+/wIBIDAGAwQAwJ
OoMAkDBAHGOgICARgwCQMEAcwCHgIBGDAGAwQA0RgBMAYDBADRGAUwBgMEANEYCTBLBAI
AAjBFMAwDBwEgAQQYFE4CAUAwCQMHACABBnwgjDAJAwcAIAEHKBgIMAkDBwAmB/rgAkUw
CQMHACoOskAAADAJAwcAKg6yQAEYBCDVgBpTRcCqvEdOUPi7RvmGw9gjloOw3NcNAwoUR
IMRAqOBmTCBljByMA0CAVAwCAICDRwCAhk9MAkCAgCuMAMCAQAwDgICAQswCAICL2ECAj
cXMB4CAgIpMBgCAgCuAgICLwICAqgCAgUTAgILYgICDPgwJgICAi8wIAICAK4CAgIBAgI
CKQICBRMCAgy5AgINHAICUeUCAlNIBCDITE9K2lIl7SnJRAygGi8R4SNruWCIgolWRu/4
xhoJjaRSMFAwLAQUE9TyT5qfzZjbNvkwYxgIyI85dLwEFOhVKx/W0aT35ATG2OVoDR68F
j/DBCCh5sjSpR+H93+2tYuqk5GZkBARAKhhAP7h+HKGR+agDKWCARkwggEVMIHwMIHtAg
I8yjCB5jBxBBRdQlDi2B1ESNiinvzpHSn/B17J4jBZMBMGByqGSM49AgEGCCqGSM49AwE
HA0IABIBXI0P4P/ywEHqwB9jKafhrnKAwBgW4SKg998DT7F8ZwBm/prWe10K1TvQ0OlJQ
EobYoOfkHxCqU7RYIqn4gBUwcQQUvoibVdC3Nzl9dcSfSFuFj6mK0R8wWTATBgcqhkjOP

```
QIBBggqhkjOPQMBBwNCAATgXEmvSfZu7HW5fUS+X5BbBli8hp0+Mu4VfabGoq4AZSEqev
tUssOCsT76X2nl4faRZM1UA3bYVRTd1v9EqkTbBCC6X7RJzvtroA82Enliou6m6Gf+hRK
73a3pxuS4vBbB0g==
```


It decodes as follows:

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

File:                       example.ccr
Hash identifier:            u8u0JbdDaij8cplt6kTaIyQFSzvgexIKuEsLhBzGhQI=
CCR produced at:            Sat 11 Apr 2026 08:04:31 +0000
Manifest state hash:        8bXskzbWa1oCoQYF1VnbQskxegvOO2eyS67YnkY29wg=
Manifest last update:       Sat 11 Apr 2026 08:00:03 +0000
Manifest instances:
                            hash:AAA2wRwPsxllQz3CGSuUSNg95LD7ve8TkQG8oJf\
Zf/Q= size:1998 aki:46387C56B331FF84BC10D8AC90E1E2C16F172345 seqnum:\
18B2 thisupdate:1775862111 sia:rsync://rpki.ripe.net/repository/DEFA\
ULT/48/1b40ff-b1e1-4951-9165-23bb39a83481/1/Rjh8VrMx_4S8ENiskOHiwW8X\
I0U.mft
                            hash:AAFxGHgJjLarAoLN6aV4ByTazpqHNrQ4xDjc5eX\
RQrY= size:2360 aki:C0D733E05D4C056E3A7E94332DC46BE80148688A seqnum:\
010D0C9F43285843EC2B3B6AE919C88C87F39200 thisupdate:1775858403 sia:r\
sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
-2171da2157d3/871da40f-793a-4a45-a0a9-978148321a07/a120b5d4-da56-49b\
1-8ae1-7987a6afde99/a120b5d4-da56-49b1-8ae1-7987a6afde99.mft
                            hash:AAW6NCJNhHhc/7A9Breu8dpn2xXFC71OKRA9u9x\
KsZQ= size:2360 aki:AE122805FDB09BF5B1DFDC14985AC40F07F43E85 seqnum:\
010D0C9F4328584B9CE67A3ADA7D16609B932F8F thisupdate:1775876403 sia:r\
sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
-2171da2157d3/85e06eaf-37a7-4588-abea-e4909ef1b4e2/b2c58ba9-3393-4f4\
c-abe1-b231a3ea2700/b2c58ba9-3393-4f4c-abe1-b231a3ea2700.mft
                            hash:AAa3zi7lmCVevHrdjC4RltPtW7yVv4w8/6KfKt+\
1EQY= size:1998 aki:16B198B6EE3AF68DEBC2347A5E98F711FB30F28B seqnum:\
05B6 thisupdate:1775876533 sia:rsync://rpki.ripe.net/repository/DEFA\
ULT/3a/22ae14-e45e-4eee-bd07-4482ada232e3/1/FrGYtu469o3rwjR6Xpj3Efsw\
8os.mft
                            hash:AAdjp/1ti4quGz7ucd/EZ/g28pqgrdsIN2FoHsI\
/O/s= size:2443 aki:02725494FA7C81427EC59F0713F3B9B068EBC8C2 seqnum:\
010D0C9F43285842762E5CDF487448DBE3D65306 thisupdate:1775862003 sia:r\
sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
-2171da2157d3/5b7fb122-dfdf-4c0c-b90d-3bc7a5feb82b/fdc3365a-18ea-469\
6-ad8c-6b66a3e152b7/fdc3365a-18ea-4696-ad8c-6b66a3e152b7.mft
                            hash:AAgBgdsAF5T25s1DMzpv6dmVkk/F8ye6gUxqw2Q\
+UWs= size:1924 aki:38D63C5FCE1EF09E4BF2CFC94BB2509FD5FF509C seqnum:\
06E4 thisupdate:1775887279 sia:rsync://rpki.ripe.net/repository/DEFA\
ULT/bf/ee3d73-9729-4da4-8bc7-67c442d6a850/1/ONY8X84e8J5L8s_JS7JQn9X_\
UJw.mft
                            hash:AAgGTG8qcTnrJ+s3v2qwsYhicint7+AAzZvEBKS\
```

```
   0/Cg= size:2072 aki:73E157B2918CADCA8A5A9FBC66E977608A6DF5E1 seqnum:\
   18BB thisupdate:1775880020 sia:rsync://rpki.ripe.net/repository/DEFA\
   ULT/79/bbcd53-c4f8-4245-bb90-00a154b8ecb1/1/c-FXspGMrcqKWp-8Zul3YIpt\
   9eE.mft
                           hash:AAxlL+UgVK9Dx4hdFmtC1Io4cnNXa+WSktfxHND\
   M2dQ= size:2360 aki:9E6EF4051C0BC6DB084757A8A37A7D5929AED033 seqnum:\
   010D0C9F43285847D8BF5EF10E9318A0E5C72573 thisupdate:1775844003 sia:r\
   sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
   -2171da2157d3/69fd0156-bb1f-48b6-bf32-c9492286f195/afb3511b-f1b6-44a\
   d-9c36-f6576a5400df/afb3511b-f1b6-44ad-9c36-f6576a5400df.mft
                           hash:ABHMuie2PQZfMvLKOxKMHfjD1eaPqk9g7ApjxVq\
   Reko= size:1998 aki:DA77FA100D6CD288E544C7CB4C7D4D18879079ED seqnum:\
   0344 thisupdate:1775865660 sia:rsync://rpki.ripe.net/repository/DEFA\
   ULT/dc/e31ad5-ef74-40a4-9a0d-df872fb269cf/1/2nf6EA1s0ojlRMfLTH1NGIeQ\
   ee0.mft
                           hash:ABRkjCfKzkq+VljmLDPOTrRzb9XCSHS4+MdJ85h\
   48Ck= size:2360 aki:4E428C958C2E77BF29924C4307CA4C7AAFCCAC8F seqnum:\
   010D0C9F432858457C8ED47A37E56E1C2B792621 thisupdate:1775894403 sia:r\
   sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
   -2171da2157d3/4ab7ae4d-bd7b-4b33-9a88-5b22d2a8337d/022d0269-7d56-45c\
   c-8167-b1e7ab13f1f4/022d0269-7d56-45cc-8167-b1e7ab13f1f4.mft
                           hash:ABmkMrkqDHq5RUqsoySxusyf+4z3PHRJyCZKs08\
   B5xw= size:2280 aki:D6BA7E3355B5CBF6740392364CA921379241C027 seqnum:\
   010D0C9F432858410FF677C49D023E85FBC3671F thisupdate:1775833203 sia:r\
   sync://rpki.arin.net/repository/arin-rpki-ta/5e4a23ea-e80a-403e-b08c\
   -2171da2157d3/2a246947-2d62-4a6c-ba05-87187f0099b2/0a94b460-7441-495\
   e-9358-475ea795ecc6/0a94b460-7441-495e-9358-475ea795ecc6.mft
   ROA payload state hash: 1YAaU0XAqrxHTlD4u0b5hsPYI5aDsNzXDQMKFESDEQI=
   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
                           94.142.240.0/21 AS 8283
                           94.142.240.0/24 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
                           185.52.224.0/22 AS 8283
                           185.52.224.0/24 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:yExPStpSJe0pyUQMoBovEeEja7lgiIKJVkbv+MYaCY0=
   ASPA payload entries:
                           customer: 80 providers: 3356, 6461
                           customer: 174 providers: 0
                           customer: 267 providers: 12129, 14103
                           customer: 553 providers: 174, 559, 680, 1299\
   , 2914, 3320
                           customer: 559 providers: 174, 513, 553, 1299\
   , 3257, 3356, 20965, 21320
   Trust anchor state hash:oebI0qUfh/d/trWLqpORmZAQEQCoYQD+4fhyhkfmoAw=
   Trust anchor keyids:    13D4F24F9A9FCD98DB36F930631808C88F3974BC, E8\
   552B1FD6D1A4F7E404C6D8E5680D1EBC163FC3
   Router key state hash:  ul+0Sc77a6APNhJ5YqLupuhn/oUSu92t6cbkuLwWwdI=
   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, deserializer, and CRDT effector implementation based on [rpkitouch] was provided by Job Snijders.

* A CCR encoding and decoding implementation in Java library [rpki-commons] was provided by RIPE NCC.

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