SIDROPS
Internet-Draft
Updates: 8630 (if approved)
Intended status: Standards Track
Expires: 16 August 2025

J. Snijders
T. Buehler
OpenBSD
T. de Kock
RIPE NCC

12 February 2025

# Tiebreaking Resource Public Key Infrastructure (RPKI) Trust Anchors
draft-ietf-sidrops-rpki-ta-tiebreaker-01

## Abstract

A Trust Anchor (TA) in the RPKI is represented by a self-signed X.509 Certification Authority (CA) certificate. Over time, Relying Parties (RP) may have acquired multiple different issuances of valid TA certificates from the same TA operator. This document proposes a tiebreaking scheme to be used by RPs to select one TA certificate for certification path validation. This document updates RFC 8630.

## 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 16 August 2025.

## Copyright Notice

Table of Contents



## 1. Introduction

In the Resource Public Key Infrastructure (RPKI) hierarchical structure, a Trust Anchor is an authority for which trust is assumed and not derived. TA operators periodically reissue TA certificates to introduce changes to, for example, modify the set of IP Address Delegations and/or Autonomous System Identifier Delegations included in the [RFC3779] extension(s), the content of the Subject Information Access extension (Section 4.2.2.2 of [RFC5280]), and the certificate validity period (Section 4.1.2.5 of [RFC5280]).

Relying Parties periodically fetch Trust Anchor certificates from well-known, remote locations and verify that the key of the self-signed certificate matches the key embedded in its associated Trust Anchor Locator [RFC8630]. This transfer may happen via an unauthenticated channel, and the certificate is verified by checking that it is signed by the public key in the TAL. After retrieving a TA certificate Relying Parties have a choice between using a previously retrieved locally cached copy of the TA certificate and the newly-retrieved instance of the TA certificate.

Periodic reissuance of TA certificates is a way of ensuring that the RPKI remains healthy at its root by avoiding ossification and retaining agility, consequently RPs refetch the certificates to adopt changes in the TA's INR [RFC3779] and SIA [RFC5280] extensions. In the past, some Trust Anchor certificates were issued with unreasonably long validity periods, in some cases up to a century. Since TA certificates are the root, and thus have no CRL covering them, Trust Anchor operators cannot revoke previously issued TA

certificates. This means that an on-path adversary or caching network element could present Relying Parties with an older instance of the TA certificate than the TA operator intends Relying Parties to use.

This document proposes a tiebreaking scheme for Relying Parties, preferring (1) the 'more recently' issued certificate, and (2) the certificate with the shortest validity period among certificates with equal notBefore. This establishes a partial order over TA certificates issued by the same TA, permitting the issuance of a certificate that is preferred over any previous certificate.

### 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.

### 1.2. Related Work

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 "A Profile for Resource Certificate Repository Structure" [RFC6481].

## 2. Updates to RFC 8630

This section updates [RFC8630].

* In Section 3, this paragraph is replaced as follows.

  OLD

  | 1. Retrieve the object referenced by (one of) the TA URI(s) contained in the TAL.
  |
  | 2. Confirm that the retrieved object is a current, self-signed RPKI CA certificate that conforms to the profile as specified in [RFC6487].
  |
  | 3. Confirm that the public key in the TAL matches the public key in the retrieved object.
  |
  | 4. Perform other checks, as deemed appropriate (locally), to ensure that the RP is willing to accept the entity publishing this self-signed CA certificate to be a TA.

| These tests apply to the validity of attestations made in
| the context of the RPKI relating to all resources described
| in the INR extension(s) of this certificate.

NEW

| 1. Retrieve the object referenced by (one of) the TA URI(s)
| contained in the TAL. If this step fails, use the locally
| cached copy of the TA referenced by the TAL previously
| retrieved.
|
| 2. Confirm that the retrieved object is a current, validly
| self-signed RPKI CA certificate that conforms to the
| profile as specified in [RFC6487]. If this step fails, use
| the locally cached copy of the retrieved TA.
|
| 3. Confirm that the public key in the TAL matches the public
| key in the retrieved object. If this step fails, use the
| locally cached copy of the retrieved TA.
|
| 4. Check whether the retrieved object has a more recent
| notBefore than the locally cached copy of the retrieved TA.
| If the notBefore of the retrieved object is less recent,
| use the locally cached copy of the retrieved TA.
|
| 5. If the notBefore dates are equal, check whether the
| retrieved object has a shorter validity period than the
| locally cached copy of the retrieved TA. If the validity
| period of the retrieved object is longer, use the locally
| cached copy of the retrieved TA.
|
| 6. If the validity period is equal, and the newly-retrieved
| certificate differs from the cached copy, use the newly-
| retrieved certificate.

## 3. Security Considerations

When Relying Parties inadvertently use a different instance of the TA certificate than the TA operator intended for RPs to use, the certification path validation process will yield an unexpected outcome. Some examples of unexpected outcomes are validation failures, or replay attacks. Standardization of a tiebreaking scheme helps both RP and TA operators arrive at deterministic outcomes. The proposed tiebreaking scheme prevents RPs from accepting a previous certificate presented by an on-path adversary in the presence of other TA certificate material.

## 4. IANA Considerations

This document has no IANA actions.

## 5. References

### 5.1. Normative References

[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>.

[RFC3779] Lynn, C., Kent, S., and K. Seo, "X.509 Extensions for IP Addresses and AS Identifiers", RFC 3779, DOI 10.17487/RFC3779, June 2004, <https://www.rfc-editor.org/info/rfc3779>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[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>.

[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>.

[RFC8630] Huston, G., Weiler, S., Michaelson, G., Kent, S., and T. Bruijnzeels, "Resource Public Key Infrastructure (RPKI) Trust Anchor Locator", RFC 8630, DOI 10.17487/RFC8630, August 2019, <https://www.rfc-editor.org/info/rfc8630>.

### 5.2. Informative References

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <https://www.rfc-editor.org/info/rfc7942>.

[rpki-client] Jeker, C., Snijders, J., Dzonsons, K., and T. Buehler, "rpki-client 9.1", June 2024, <https://www.rpki-client.org/>.

[rpki-prover] Puzanov, M., "rpki-prover", August 2024, <https://github.com/lolepezy/rpki-prover/pull/218>.

## Appendix A. 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 [RFC7942]. 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 [RFC7942], "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".

* OpenBSD [rpki-client]

* Mikhail Puzanov’s [rpki-prover]

## Acknowledgements

The authors wish to thank Tim Bruijnzeels and George Michaelson.

## Authors’ Addresses

Job Snijders
Amsterdam
The Netherlands
Email: job@sobornost.net

Theo Buehler
OpenBSD
Switzerland
Email: tb@openbsd.org

Ties de Kock
RIPE NCC
Amsterdam
Netherlands
Email: tdekock@ripe.net