Stream:	Internet Engineering Task Force (IETF)		
RFC:	10097		
Updates:	8630		
Category:	Standards Track		
Published:	November 2026		
ISSN:	2070-1721		
Authors:	J. Snijders	T. Buehler	T. de Kock
	BSD	*OpenBSD*	*RIPE NCC*

RFC 10097
Tiebreaking Resource Public Key Infrastructure (RPKI) Trust Anchors

Abstract

A Trust Anchor (TA) in the Resource Public Key Infrastructure (RPKI) is represented by a self-signed X.509 Certification Authority (CA) certificate. Over time, Relying Parties (RPs) may have acquired multiple different issuances of valid TA certificates from the same TA operator. This document specifies 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 is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc10097.

Copyright Notice

Copyright (c) 2026 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions

with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction 2
   1.1. Requirements Language 3
   1.2. Related Work 3
2. Updates to RFC 8630 3
3. Security Considerations 4
4. IANA Considerations 4
5. Normative References 5
Acknowledgements 5
Authors' Addresses 5

1. Introduction

In the Resource Public Key Infrastructure (RPKI) hierarchical structure, a Trust Anchor (TA) is an authority for which trust is assumed and not derived. TA operators periodically reissue TA certificates to update the validity period (Section 4.1.2.5 of [RFC5280]), the Subject Information Access (SIA) extension (Section 4.2.2.2 of [RFC5280]), the Certificate Policies extension (Section 4.2.1.4 of [RFC5280]), and the Internet Number Resources (INRs) ([RFC3779]).

Relying Parties (RPs) periodically fetch TA certificates from online locations and verify that the key of the self-signed certificate matches the key embedded in its associated Trust Anchor Locator (TAL) [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, RPs have a choice between using a previously retrieved locally cached copy of the TA certificate and the newly retrieved instance of the TA certificate, provided that both certificates are valid.

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 TA 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 Certificate Revocation List (CRL) [RFC5280]

covering their own scope, TA operators cannot revoke previously issued TA certificates. This means that an on-path adversary or a caching network element could present RPs with an older instance of the TA certificate than the TA operator intends RPs to use.

This document specifies a tiebreaking scheme for RPs, preferring:

(1) the "more recently" issued TA certificate,

(2) the TA certificate with the shortest validity period among certificates with equal notBefore (Section 4.6.1 of [RFC6487]), and

(3) the "most recently fetched" instance of the TA certificate among certificates with equal notBefore and equal notAfter (Section 4.6.2 of [RFC6487]).

This establishes a preorder over TA certificates issued by the same TA, permitting the issuance of a certificate that is preferred over any previous certificate.

This document updates Section 3 of [RFC8630].

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

The following text in Section 3 of [RFC8630] is replaced as follows:

OLD:

> In order to use the TAL to retrieve and validate a (putative) TA, an RP **SHOULD**:
>
> 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:

In order to use the TAL to retrieve and validate a (putative) TA, an RP **MUST**:

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 newly 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 TA.
3. Confirm that the public key in the TAL matches the public key in the newly retrieved putative TA. If the keys do not match, use the locally cached TA. If the keys match and there is no locally cached TA, use the newly retrieved TA.
4. Check whether the newly retrieved TA has a more recent `notBefore` than the locally cached TA. If the `notBefore` of the newly retrieved TA is less recent, use the locally cached TA. If the `notBefore` of the newly retrieved TA is more recent, use the newly retrieved TA.
5. If the `notBefore` dates are equal, check whether the newly retrieved TA has a shorter validity period than the locally cached TA. If the validity period of the newly retrieved TA is longer, use the locally cached TA. If the validity period of the newly retrieved TA is shorter, use the newly retrieved TA.
6. If the validity periods are equal and the newly retrieved TA differs from the locally cached TA, use the newly retrieved TA. In the unlikely event that this step is reached, it seems most likely that the TA operator intended for RPs to use the TA that is currently published.

3. Security Considerations

When RPs inadvertently use a different instance of the TA certificate than that which 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 specified 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. 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>.

Acknowledgements

The authors wish to thank Tim Bruijnzeels, George Michaelson, Tom Harrison, Russ Housley, Mohamed Boucadair, Christer Holmberg, and Mahesh Jethanandani for their review and feedback.

Authors' Addresses

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

Theo Buehler
OpenBSD
Switzerland
Email: tb@openbsd.org

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