

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 12, 2017

T. King
D. Kopp
DE-CIX
A. Lambrianidis
AMS-IX
A. Fenoux
France-IX
April 10, 2017

Signaling Prefix Origin Validation Results from a Route Server to Peers
draft-ietf-sidrops-route-server-rpki-light-02

Abstract

This document defines the usage of the BGP Prefix Origin Validation State Extended Community [RFC8097] to signal prefix origin validation results from a route server to its peers. Upon reception of prefix origin validation results peers can use this information in their local routing decision process.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC2119] only when they appear in all upper case. They may also appear in lower or mixed case as English words, without normative meaning.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on October 12, 2017.

Copyright Notice

Copyright (c) 2017 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. BGP Prefix Origin Validation State Utilized at Route-Servers	3
3. Signaling Prefix Origin Validation Results from a Route Server to Peers	4
4. Operational Recommendations	4
4.1. Local Routing Decision Process	4
4.2. Route Server Receiving the BGP Prefix Origin Validation State Extended Community	4
4.3. Information about Validity of a BGP Prefix Origin Not Available at a Route-Server	5
4.4. Error Handling at Peers	5
5. IANA Considerations	5
6. Security Considerations	5
7. References	6
7.1. Normative References	6
7.2. Informative References	6
Authors' Addresses	7

1. Introduction

RPKI-based prefix origin validation [RFC6480] can be a significant operational burden for BGP peers to implement and adopt. In order to boost acceptance and usage of prefix origin validation and ultimately increase the security of the Internet routing system, IXPs may provide RPKI-based prefix origin validation at the route server [RFC7947]. The result of this prefix origin validation is signaled to peers by using the BGP Prefix Origin Validation State Extended Community as introduced in [RFC8097].

Peers receiving the prefix origin validation result from the route server(s) can use this information in their local routing decision

process for acceptance, rejection, preference, or other traffic engineering purposes of a particular route.

2. BGP Prefix Origin Validation State Utilized at Route-Servers

A route server that is aware of a BGP Prefix Origin Validation state for a certain route can handle this information in one of the following modes of operation:

Simple Tagging: The prefix origin validation state is tagged to the route as described in Section 3.

This mode of operation is like the traditional way route servers work, however, the prefix origin validation state information is additionally available for peers.

Dropping and Tagging: Routes for which the prefix origin validation state is "invalid" (according to [RFC6811]) are dropped by the route server. Routes which show a prefix origin validation state of "not found" and "valid" (according to [RFC6811]) are tagged accordingly to Section 3.

Security is higher rated than questionable reachability of a prefix by this mode of operation.

Prioritizing and Tagging: If the route server learned for a particular prefix more than one route it removes firstly the set of "invalid" routes and secondly the "not found" routes unless the set of routes is empty. Based on the set of routes left over the BGP best path section algorithm is executed. The selected route is marked accordingly to Section 3.

The BGP best path selection algorithm is changed by this mode of operation in such a way that "valid" routes are preferred even if they are unfavorable by the traditional best path selection algorithm. This puts prefix origin validation on top of the best path selection.

A route server MUST support the Simple Tagging mode of operation. Other modes of operation are OPTIONAL. The mode of operation MAY be configured by the route server operator for a route server instance or for each BGP session with a peer separately.

These mode of operations might be used in combination with [RFC7911] in order to allow a peer to receive all routes and take the routing decision by itself.

3. Signaling Prefix Origin Validation Results from a Route Server to Peers

The BGP Prefix Origin Validation State Extended Community (as defined in [RFC8097]) is utilized for signaling prefix origin validation result from a route server to peers.

[RFC8097] proposes an encoding of the prefix origin validation result [RFC6811] as follows:

Value	Meaning
0	Lookup result = "valid"
1	Lookup result = "not found"
2	Lookup result = "invalid"

Table 1

This encoding is re-used. Route servers providing RPKI-based prefix origin validation set the validation state according to the prefix origin validation result (see [RFC6811]).

4. Operational Recommendations

4.1. Local Routing Decision Process

A peer receiving prefix origin validation results from the route server MAY use the information in its own local routing decision process. The local routing decision process SHOULD apply to the rules as described in section 5 [RFC6811].

A peer receiving a prefix origin validation result from the route server MAY redistribute this information within its own AS.

4.2. Route Server Receiving the BGP Prefix Origin Validation State Extended Community

An IXP route server receiving routes from its peers containing the BGP Prefix Origin Validation State Extended Community MUST remove the extended community before the route is re-distributed to its peers. This is required regardless of whether the route server is executing prefix origin validation or not.

Failure to do so would allow opportunistic peers to advertise routes tagged with arbitrary prefix origin validation results via a route

server, influencing maliciously the decision process of other route server peers.

4.3. Information about Validity of a BGP Prefix Origin Not Available at a Route-Server

In case information about the validity of a BGP prefix origin is not available at the route server (e.g., error in the ROA cache, CPU overload) the route server MUST NOT add the BGP Prefix Origin Validation State Extended Community to the route.

4.4. Error Handling at Peers

A route sent by a route server SHOULD only contain none or one BGP Prefix Origin Validation State Extended Community.

A peer receiving a route from a route server containing more than one BGP Prefix Origin Validation State Extended Community SHOULD only consider the largest value (as described in Table 1) in the validation result field and disregard the other values. Values larger than two in the validation result field MUST be disregarded.

5. IANA Considerations

None.

6. Security Considerations

All security considerations described in RFC 6811 [RFC6811] fully apply to this document.

Additionally, threat agents polluting ROA cache server(s) run by IXP operators could cause significant operational impact, since multiple route server clients could be affected. Peers should be vigilant as to the integrity and authenticity of the origin validation results, as they are provided by a third party, namely the IXP operator hosting both the route server as well as any ROA cache server(s).

Therefore, a route server could be misused to spread malicious prefix origin validation results. However, peers already trust the route server for the collection, filtering (e.g. IRR database filtering), and redistribution of BGP routing information to other peers. So, no change in the trust level is needed for this proposal.

To facilitate trust and help with peers establishing appropriate controls in mitigating the risks mentioned above, IXPs SHOULD provide out-of-band means for peers to ensure that the ROA validation process has not been compromised or corrupted.

While being under DDoS attacks, it is a common practice for peers connected to an IXP to make use of blackholing services (see [RFC7999]). Peers are using blackholing to drop traffic, typically by announcing a more specific prefix, which is under attack. A peer SHOULD make sure that this prefix is covered by an appropriate ROA.

7. References

7.1. 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, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, DOI 10.17487/RFC4360, February 2006, <<http://www.rfc-editor.org/info/rfc4360>>.
- [RFC6811] Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", RFC 6811, DOI 10.17487/RFC6811, January 2013, <<http://www.rfc-editor.org/info/rfc6811>>.
- [RFC7911] Walton, D., Retana, A., Chen, E., and J. Scudder, "Advertisement of Multiple Paths in BGP", RFC 7911, DOI 10.17487/RFC7911, July 2016, <<http://www.rfc-editor.org/info/rfc7911>>.
- [RFC8097] Mohapatra, P., Patel, K., Scudder, J., Ward, D., and R. Bush, "BGP Prefix Origin Validation State Extended Community", RFC 8097, DOI 10.17487/RFC8097, March 2017, <<http://www.rfc-editor.org/info/rfc8097>>.

7.2. Informative References

- [RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, DOI 10.17487/RFC6480, February 2012, <<http://www.rfc-editor.org/info/rfc6480>>.
- [RFC7947] Jasinska, E., Hilliard, N., Raszuk, R., and N. Bakker, "Internet Exchange BGP Route Server", RFC 7947, DOI 10.17487/RFC7947, September 2016, <<http://www.rfc-editor.org/info/rfc7947>>.

[RFC7999] King, T., Dietzel, C., Snijders, J., Doering, G., and G.
Hankins, "BLACKHOLE Community", RFC 7999,
DOI 10.17487/RFC7999, October 2016,
<<http://www.rfc-editor.org/info/rfc7999>>.

Authors' Addresses

Thomas King
DE-CIX Management GmbH
Lichtstrasse 43i
Cologne 50825
DE

Email: thomas.king@de-cix.net

Daniel Kopp
DE-CIX Management GmbH
Lichtstrasse 43i
Cologne 50825
DE

Email: daniel.kopp@de-cix.net

Aristidis Lambrianidis
Amsterdam Internet Exchange
Frederiksplein 42
Amsterdam 1017 XN
NL

Email: aristidis.lambrianidis@ams-ix.net

Arnaud Fenioux
France-IX
88 Avenue Des Ternes
Paris 75017
FR

Email: afenioux@franceix.net

SIDROPS
Internet-Draft
Intended status: Informational
Expires: May 15, 2018

D. Ma
ZDNS
S. Kent
BBN
November 11, 2017

Requirements for Resource Public Key Infrastructure (RPKI) Relying
Parties
draft-ietf-sidrops-rp-00

Abstract

This document provides a single reference point for requirements for Relying Party (RP) software for use in the Resource Public Key Infrastructure (RPKI). It cites requirements that appear in several RPKI RFCs, making it easier for implementers to become aware of these requirements that are segmented with orthogonal functionalities.

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 May 15, 2018.

Copyright Notice

Copyright (c) 2017 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 Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Fetching and Caching RPKI Repository Objects	3
2.1. TAL Acquisition and Processing	4
2.2. Locating RPKI Objects Using Authority and Subject Information Extensions	4
2.3. Dealing with Key Rollover	4
2.4. Dealing with Algorithm Transition	4
2.5. Strategies for Efficient Cache Maintenance	5
3. Certificate and CRL Processing	5
3.1. Verifying Resource Certificate and Syntax	5
3.2. Certificate Path Validation	5
3.3. CRL Processing	5
4. Processing RPKI Repository Signed Objects	6
4.1. Basic Signed Object Syntax Checks	6
4.2. Syntax and Validation for Each Type of Signed Object . .	6
4.2.1. Manifest	6
4.2.2. ROA	6
4.2.3. Ghostbusters	7
4.2.4. Verifying BGPsec Router Certificate	7
4.3. How to Make Use of Manifest Data	7
4.4. What to Do with Ghostbusters Information	8
5. Delivering Validated Cache to BGP Speakers	8
6. Security considerations	8
7. IANA Considerations	8
8. Acknowledgements	8
9. References	8
9.1. Normative References	8
9.2. Informative References	10
Authors' Addresses	10

1. Introduction

The RPKI RP software is used by network operators and others to acquire and verify Internet Number Resource (INR) data stored in the RPKI repository system. RPKI data, when verified, allow an RP to verify assertions about which Autonomous Systems (ASes) are authorized to originate routes for IP address prefixes. RPKI data also establishes binding between public keys and BGP routers, and indicates the AS numbers that each router is authorized to represent.

Noting that the essential requirements imposed on RPs are scattered throughout numerous RFC documents that are protocol specific or provide best practices, as follows:

RFC 6481 (Repository Structure)
RFC 6482 (ROA format)
RFC 6486 (Manifests)
RFC 6487 (Certificate and CRL profile)
RFC 6488 (RPKI Signed Objects)
RFC 6489 (Key Rollover)
RFC 6810 (RPKI to Router Protocol)
RFC 6916 (Algorithm Agility)
RFC 7730 (Trust Anchor Locator)
RFC 7935 (Algorithms)
RFC 8209 (Router Certificates)

This makes it hard for an implementer to be confident that he/she has addressed all of these generalized requirements. Besides, software engineering calls for how to segment the RP system into components with orthogonal functionalities, so that those components could be distributed across the operational timeline of the user. Taxonomy of generalized RP requirements is going to help have 'RP role' well framed.

To consolidate RP requirements in one document, with pointers to all the relevant RFCs, this document outlines a set of baseline requirements imposed on RPs and provides a single reference point for requirements for RP software for use in the RPKI, as segmented with orthogonal functionalities:

- o Fetching and Caching RPKI Repository Objects
- o Processing Certificates and CRLs
- o Processing RPKI Repository Signed Objects
- o Delivering Validated Cache Data to BGP Speakers

This document will be update to reflect new or changed requirements as these RFCs are updated, or new RFCs are written.

2. Fetching and Caching RPKI Repository Objects

RP software uses synchronization mechanisms supported by targeted repositories (e.g., [rsync]) to download all RPKI changed data objects in the repository system and cache them locally. The software validates the RPKI data and uses it to generate authenticated data identifying which ASes are authorized to originate routes for address prefixes, and which routers are authorized to sign BGP updates on behalf of ASes.

2.1. TAL Acquisition and Processing

In the RPKI, each relying party (RP) chooses its own set of trust anchors (TAs). Consistent with the extant INR allocation hierarchy, the IANA and/or the five RIRs are obvious candidates to be default TAs for the RP.

An RP does not retrieve TAs directly. A set of Trust Anchor Locators (TALs) is used by each RP to retrieve and verify the authenticity of each trust anchor.

TAL acquisition and processing are specified in Section 3 of [RFC7730].

2.2. Locating RPKI Objects Using Authority and Subject Information Extensions

The RPKI repository system is a distributed one, consisting of multiple repository instances. Each repository instance contains one or more repository publication points. An RP discovers publication points using the SIA and AIA extensions from (validated) certificates.

Section 5 of [RFC6481] specifies how an RP locates all RPKI objects by using the SIA and AIA extensions. Detailed specifications of SIA and AIA extensions in a resource certificate are described in section 4 of [RFC6487].

2.3. Dealing with Key Rollover

An RP takes the key rollover period into account with regard to its frequency of synchronization with RPKI repository system.

RP requirements in dealing with key rollover are described in section 3 of [RFC6489].

2.4. Dealing with Algorithm Transition

The set of cryptographic algorithms used with the RPKI is expected to change over time. Each RP is expected to be aware of the milestones established for the algorithm transition and what actions are required at every juncture.

RP requirements for dealing with algorithm transition are specified in section 4 of [RFC6916].

2.5. Strategies for Efficient Cache Maintenance

Each RP is expected to maintain a local cache of RPKI objects. The cache needs to be as up to date and consistent with repository publication point data as the RP's frequency of checking permits.

The last paragraph of section 5 of [RFC6481] provides guidance for maintenance of a local cache.

3. Certificate and CRL Processing

The RPKI make use of X.509 certificates and CRLs, but it profiles these standard formats [RFC6487]. The major change to the profile established in [RFC5280] is the mandatory use of a new extension to X.509 certificate [RFC3779].

3.1. Verifying Resource Certificate and Syntax

Certificates in the RPKI are called resource certificates, and they are required to conform to the profile [RFC6487]. An RP is required to verify that a resource certificate adheres to the profile established by [RFC6487]. This means that all extensions mandated by [RFC6487] must be present and value of each extension must be within the range specified by this RFC. Moreover, any extension excluded by [RFC6487] must be omitted.

Section 7.1 of [RFC6487] gives the procedure that the RP should follow to verify resource certificate and syntax.

3.2. Certificate Path Validation

In the RPKI, issuer can only assign and/or allocate public INRs belong to it, thus the INRs in issuer's certificate are required to encompass the INRs in the subject's certificate. This is one of necessary principles of certificate path validation in addition to cryptographic verification i.e., verification of the signature on each certificate using the public key of the parent certificate).

Section 7.2 of [RFC6487] gives the procedure that the RP should follow to perform certificate path validation.

3.3. CRL Processing

The CRL processing requirements imposed on CAs and RP are described in [RFC6487]. CRLs in the RPKI are tightly constrained; only the AuthorityKeyIdentifier and CRLNumber extensions are allowed, and they MUST be present. No other CRL extensions are allowed, and no CRLentry extensions are permitted. RPs are required to verify that

these constraints have been met. Each CRL in the RPI MUST be verified using the public key from the certificate of the CA that issued the CRL.

In the RPKI, RPs are expected to pay extra attention when dealing with a CRL that is not consistent with the Manifest associated with the publication point associated with the CRL.

Processing of a CRL that is not consistent with a manifest is a matter of local policy, as described in the fourth paragraph of Section 6.6 of [RFC6486].

4. Processing RPKI Repository Signed Objects

4.1. Basic Signed Object Syntax Checks

Before an RP can use a signed object from the RPKI repository, the RP is required to check the signed object syntax.

Section 3 of [RFC6488] lists all the steps that the RP is required to execute in order to validate the top level syntax of a repository signed object.

Note that these checks are necessary, but not sufficient. Additional validation checks must be performed based on the specific type of signed object.

4.2. Syntax and Validation for Each Type of Signed Object

4.2.1. Manifest

To determine whether a manifest is valid, the RP is required to perform manifest-specific checks in addition to those specified in [RFC6488].

Specific checks for a Manifest are described in section 4 of [RFC6486]. If any of these checks fails, indicating that the manifest is invalid, then the manifest will be discarded and treated as though no manifest were present.

4.2.2. ROA

To validate a ROA, the RP is required to perform all the checks specified in [RFC6488] as well as the additional ROA-specific validation steps. The IP address delegation extension [RFC3779] present in the end-entity (EE) certificate (contained within the ROA), must encompass each of the IP address prefix(es) in the ROA.

More details for ROA validation are specified in section 2 of [RFC6482].

4.2.3. Ghostbusters

The Ghostbusters Record is optional; a publication point in the RPKI can have zero or more associated Ghostbuster Records. If a CA has at least one Ghostbuster Record, RP is required to verify that this Ghostbusters Record conforms to the syntax of signed object defined in [RFC6488].

The payload of this signed object is a (severely) profiled vCard. An RP is required to verify that the payload of Ghostbusters conforms to format as profiled in [RFC6493].

4.2.4. Verifying BGPsec Router Certificate

A BGPsec Router Certificate is a resource certificate, so it is required to comply with [RFC6487]. Additionally, the certificate must contain an AS Identifier Delegation extension, and must not contain an IP Address Delegation extension. The validation procedure used for BGPsec Router Certificates is identical to the validation procedure described in Section 7 of [RFC6487], but using the constraints applied come from specification of section 7 of [RFC8209].

Note that the cryptographic algorithms used by BGPsec routers are found in [RFC8208]. Currently, the algorithms specified in [RFC8208] and [RFC7935] are different. BGPsec RPs will need to support algorithms that are used to validate BGPsec signatures as well as the algorithms that are needed to validate signatures on BGPsec certificates, RPKI CA certificates, and RPKI CRLs.

4.3. How to Make Use of Manifest Data

For a given publication point, the RP ought to perform tests to determine the state of the Manifest at the publication point. A Manifest can be classified as either valid or invalid, and a valid Manifest is either current and stale. An RP decides how to make use of a Manifest based on its state, according to local (RP) policy.

If there are valid objects in a publication point that are not present on a Manifest, [RFC6486] does not mandate specific RP behavior with respect to such objects. However, most RP software ignores such objects and this document recommends that this behavior be adopted uniformly.

In the absence of a Manifest, an RP is expected to accept all valid signed objects present in the publication point. If a Manifest is stale (see [RFC6486]) and an RP has no way to acquire a more recent Manifest, the RP is expected to (TBD).

4.4. What to Do with Ghostbusters Information

An RP may encounter a stale Manifest or CRL, or an expired CA certificate or ROA at a publication point. An RP is expected to use the information from the Ghostbusters record to contact the maintainer of the publication point where any stale/expired objects were encountered. The intent here is to encourage the relevant CA and/or repository manager to update the slate or expired objects.

5. Delivering Validated Cache to BGP Speakers

On a periodic basis, BGP speakers within an AS request updated validated origin AS data and router/ASN data from the RP's cache. The RP passes this information to BGP speakers to enable them to verify the authenticity of routing announcements. The specification of the protocol designed to deliver validated cache data from an RP to a BGP Speaker is provided in [RFC6810].

6. Security considerations

TBD

7. IANA Considerations

This document has no actions for IANA.

8. Acknowledgements

The authors thank David Mandelberg, Wei Wang and Tim Bruijnzeels for their review, feedback and editorial assistance in preparing this document.

9. References

9.1. Normative References

[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>>.
- [RFC6482] Lepinski, M., Kent, S., and D. Kong, "A Profile for Route Origin Authorizations (ROAs)", RFC 6482, DOI 10.17487/RFC6482, February 2012, <<https://www.rfc-editor.org/info/rfc6482>>.
- [RFC6486] Austein, R., Huston, G., Kent, S., and M. Lepinski, "Manifests for the Resource Public Key Infrastructure (RPKI)", RFC 6486, DOI 10.17487/RFC6486, February 2012, <<https://www.rfc-editor.org/info/rfc6486>>.
- [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>>.
- [RFC6489] Huston, G., Michaelson, G., and S. Kent, "Certification Authority (CA) Key Rollover in the Resource Public Key Infrastructure (RPKI)", BCP 174, RFC 6489, DOI 10.17487/RFC6489, February 2012, <<https://www.rfc-editor.org/info/rfc6489>>.
- [RFC6493] Bush, R., "The Resource Public Key Infrastructure (RPKI) Ghostbusters Record", RFC 6493, DOI 10.17487/RFC6493, February 2012, <<https://www.rfc-editor.org/info/rfc6493>>.
- [RFC6810] Bush, R. and R. Austein, "The Resource Public Key Infrastructure (RPKI) to Router Protocol", RFC 6810, DOI 10.17487/RFC6810, January 2013, <<https://www.rfc-editor.org/info/rfc6810>>.

- [RFC6916] Gagliano, R., Kent, S., and S. Turner, "Algorithm Agility Procedure for the Resource Public Key Infrastructure (RPKI)", BCP 182, RFC 6916, DOI 10.17487/RFC6916, April 2013, <<https://www.rfc-editor.org/info/rfc6916>>.
- [RFC7730] Huston, G., Weiler, S., Michaelson, G., and S. Kent, "Resource Public Key Infrastructure (RPKI) Trust Anchor Locator", RFC 7730, DOI 10.17487/RFC7730, January 2016, <<https://www.rfc-editor.org/info/rfc7730>>.
- [RFC7935] Huston, G. and G. Michaelson, Ed., "The Profile for Algorithms and Key Sizes for Use in the Resource Public Key Infrastructure", RFC 7935, DOI 10.17487/RFC7935, August 2016, <<https://www.rfc-editor.org/info/rfc7935>>.
- [RFC8208] Turner, S. and O. Borchert, "BGPsec Algorithms, Key Formats, and Signature Formats", RFC 8208, DOI 10.17487/RFC8208, September 2017, <<https://www.rfc-editor.org/info/rfc8208>>.
- [RFC8209] Reynolds, M., Turner, S., and S. Kent, "A Profile for BGPsec Router Certificates, Certificate Revocation Lists, and Certification Requests", RFC 8209, DOI 10.17487/RFC8209, September 2017, <<https://www.rfc-editor.org/info/rfc8209>>.

9.2. Informative References

- [RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, DOI 10.17487/RFC6480, February 2012, <<https://www.rfc-editor.org/info/rfc6480>>.
- [rsync] "rsync web page", <<http://rsync.samba.org/>>.

Authors' Addresses

Di Ma
ZDNS
4 South 4th St. Zhongguancun
Haidian, Beijing 100190
China

Email: madi@zdns.cn

Stephen Kent
BBN
10 Moulton St
Cambridge, MA 02138-1119
USA

Email: kent@alum.mit.edu

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 17, 2018

T. Bruijnzeels
RIPE NCC
C. Martinez
LACNIC
November 13, 2017

RPKI signed object for TAL
draft-ietf-sidrops-signed-tal-00

Abstract

Trust Anchor Locators (TALs) [RFC7730] are used by Relying Parties in the RPKI to locate and validate Trust Anchor certificates used in RPKI validation. This document defines an RPKI signed object [RFC6488] for a Trust Anchor Locator (TAL) that can be published by Trust Anchor to communicate a new TAL to already deployed Relying Parties. The two primary use cases for this are that 1) a Trust Anchor may wish to change the locations where its TA certificate may be found, and 2) a Trust Anchor may wish to perform a planned migration to a new key. Note that unplanned key rolls are considered out of scope for this document.

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 May 17, 2018.

Copyright Notice

Copyright (c) 2017 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Requirements notation	2
2. Introduction	3
3. Signed TAL definition	3
3.1. The Signed TAL Content Type	4
3.2. The Signed TAL eContent	4
3.3. Signed TAL Validation	4
4. Signed TAL Generation	4
5. Signed TAL Publication	5
6. Supporting a TA Key Roll	5
6.1. Preparing a new TA key	6
6.2. Staging period - Using both the old and the new TA key	6
6.3. Preserving the Signed TAL	6
6.4. Retiring the old key	7
6.5. Relying Party Use	7
7. Supporting changing TA certificate publication point(s)	7
7.1. Adding a publication point	7
7.2. Withdrawing a publication point	7
7.3. Publishing the Signed TAL	7
7.4. Relying Party Use	7
8. IANA Considerations	8
8.1. OID	8
8.2. File Extension	8
9. Security Considerations	8
10. Acknowledgements	8
11. References	8
11.1. Normative References	8
11.2. Informative References	9
Authors' Addresses	10

1. Requirements notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Introduction

Trust Anchor Locator (TAL) files [RFC7730] are used in the Resource Public Key Infrastructure (RPKI) to help Relying Parties locate and verify a trust anchor certificate. A TAL file consists of:

- o One or more rsync URIs [RFC5781]
- o A subjectPublicKeyInfo [RFC5280] in DER format [X.509], encoded in Base64

The TAL can be distributed out-of-band to Relying Parties (RP), and it allows the RP to retrieve the most recent version of the Trust Anchor (TA) certificate from the cited location, and verify that public key of this certificate matches the TAL. This is useful as it allows selected data in the trust anchor to change, without needing to effect redistribution of the trust anchor per se. In particular the Internet Number Resources (INRs) extension [RFC3779] and the publication points defined in the Subject Information Access [RFC6487] may be updated this way.

The assumption is that both the URIs and key of the TA certificate remain stable. However, an organisation operating a TA may wish to change either of these properties, because of a need to:

- o change one or more URIs
- o perform a planned key roll

In this document we describe a method for TA operators to publish a an updated TAL in a secure a well-defined fashion, so that RPs can be alerted about these changes.

Note that [RFC5011] describes Automated Updates of DNS Security (DNSSEC) Trust Anchors and can provide some useful insight here as well. However, concepts like a set of Trust Anchors, standby Trust Anchors, and TTLs are not applicable to the RPKI. Therefore we believe that an alternative approach based on already existing concept of the Trust Anchor Locator [RFC7730] is appropriate.

3. Signed TAL definition

A signed TAL is an RPKI signed object, as specified in [RFC6488]. The RPKI signed object template requires specification of the following data elements in the context of the manifest structure.

3.1. The Signed TAL Content Type

This document requests an OID for signed-Tal as follows:

```
signed-Tal OBJECT IDENTIFIER ::= { iso(1) member-body(2) us(840)
    rsadsi(113549) pkcs(1) pkcs9(9) 16 id-smime (1) TBD }
```

This OID MUST appear both within the eContentType in the encapContentInfo object as well as the content-type signed attribute in the signerInfo object (see [RFC6488]).

3.2. The Signed TAL eContent

The content of a Signed TAL is ASN.1 encoded using the Distinguished Encoding Rules (DER) [X.690], and is defined as follows:

```
SignedTalContent ::= IA5String
```

The "SignedTalContent" contains the content of the new TAL encoded in Base64 [RFC4648].

3.3. Signed TAL Validation

Before a Relying Party can use a Signed TAL, the relying party MUST first validate the Signed TAL. To validate a Signed TAL, the relying party MUST perform all the validation checks specified in [RFC6488] as well as the following additional specific validation step.

- o The eContentType in the EncapsulatedContentInfo has OID 1.2.840.113549.1.9.16.1.TBD.
- o The EE certificate of this Signed TAL is signed by a known Trust Anchor
- o The decoded TAL content conforms to the format defined in [RFC7730]

If the above procedure indicates that the manifest is invalid, then the Signed TAL MUST be discarded and treated as though no Signed TAL were present.

4. Signed TAL Generation

A TA MAY choose to generate a single Signed TAL object to publish in its TA certificate publication point(s) in the RPKI. The TA MUST perform the following steps to generate the Signed TAL:

- o Generate a key pair for a "one-time-use" EE certificate to use for the Signed TAL
- o Generate a one-time-use EE certificate for the Signed TAL
- o This EE certificate MUST have an SIA extension access description field with an accessMethod OID value of id-ad-signedobject, where the associated accessLocation references the publication point of the Signed TAL as an object URL.
- o As described in [RFC6487], an [RFC3779] extension is required in the EE certificate used for this object. However, because the resource set is irrelevant to this object type, this certificate MUST describe its Internet Number Resources (INRs) using the "inherit" attribute, rather than explicit description of a resource set.
- o This EE certificate MUST have a "notBefore" time that is before the moment that the Signed TAL will be published.
- o This EE certificate MUST have a "notAfter" time that reflects the intended time that this Signed TAL will be published. If the EE certificate for a Signed TAL is expired, it MUST no longer be published, but of course it MAY be replaced by a newly generated Signed TAL object with similar content and an updated "notAfter" time.

5. Signed TAL Publication

A TA MAY publish a single Signed TAL object directly under its CA repository publication points. A non-normative guideline for naming this object is that the filename chosen for the signed TAL in the publication repository be a value derived from the public key part of the entity's key pair, using the algorithm described for CRLs in section 2.2 of [RFC6481] for generation of filenames. The filename extension of ".tal" MUST be used to denote the object as a signed TAL. Note that this is in-line with filename extensions defined in section 7.2 of [RFC6481].

6. Supporting a TA Key Roll

A Signed TAL MAY be used to communicate a planned key roll for the TA.

6.1. Preparing a new TA key

Prior to publishing the Signed TAL for the new key the TA MUST perform the following steps:

- o Generate a new key pair for the new TA certificate
- o Generate a new TA Certificate, using a Subject Information Access for CA certificates (see section 4.8.8.1 of [RFC6487]) that references the URIs that will be used by the new key to publish objects, that are different from the URIs used by the TA certificate for the current key.
- o ALL current signed certificates and other objects, with the exception of the old CRL, Manifest and Signed TAL, must be re-issued by the new key and published under the new publication point(s).
- o The new TA certificate itself MUST be published in a (number of) new location(s) that are different from where the TA certificate for the current key is published.

After these steps are performed a new Signed TAL MUST be generated as described in Section 4, and published as described in Section 5.

6.2. Staging period - Using both the old and the new TA key

The staging period is initiated by the initial publication of a Signed TAL for the new key and must be last at least 24 HOURS. During the staging period the TA MUST continue to operate both the old and the new TA key. Note that this is the same staging period used for key roll of normal CAs in the RPKI, described in [RFC6489].

6.3. Preserving the Signed TAL

The TA SHOULD preserve a Signed TAL for the old key after the staging period as a hint for RPs that missed the key roll. The following process can be used to achieve this:

- o Produce a new long-lived CRL that revokes all previously signed certificates
- o Produce a new long-lived Signed TAL
- o Produce a new long-lived manifest that includes the CRL and Signed TAL
- o Publish the CRL, MFT and Signed TAL

- o Destroy the old TA key

6.4. Retiring the old key

The TA SHOULD retire and delete its old key after the staging period is over.

6.5. Relying Party Use

When an RP discovers a valid Signed TAL signed under a TA, and it notices that the contained TAL is different from its current TAL for this TA and that the "subjectPublicKeyInfo" has changed, then the RP MUST replace the TAL for this TA with the new TAL, abort the current top-down validation operation, and initiate a new top-down validation operation using the updated TAL.

It is RECOMMENDED that the software informs the operator of this event.

7. Supporting changing TA certificate publication point(s)

A signed TAL MAY be used to communicate an addition or removal of one or more publication locations where the TA certificate can be found.

7.1. Adding a publication point

When adding a publication point for a TA certificate, the TA MUST publish the certificate in the new location(s) prior to publication of the Signed TAL.

7.2. Withdrawing a publication point

When removing a publication point for TA certificate, the TA SHOULD observe a staging period of at least 24 Hours. The staging period is initiated by the publication of an updated Signed TAL where the publication point has been removed. During the staging period the TA SHOULD keep the old publication point up to date and available.

7.3. Publishing the Signed TAL

It is RECOMMENDED that a Trust Anchor publishes a valid Signed TAL for what it believes its current TAL should be at all times.

7.4. Relying Party Use

When an RP discovers a valid Signed TAL signed under a TA, and it notices that the contained TAL is different from its current TAL for this TA and that the "subjectPublicKeyInfo" has not changed, then the

RP MUST replace the TAL for this TA with the new TAL for future use, but can continue the current top-down validation operation.

It is RECOMMENDED that the software informs the operator of this event.

8. IANA Considerations

8.1. OID

IANA is to add the following to the "RPKI Signed Objects" registry:

Decimal	Description	References
TBD	signed-Tal	[section 3.1]

8.2. File Extension

IANA is to add an item for the Signed TAL file extension to the "RPKI Repository Name Scheme" created by [RFC6481] as follows:

Extension	RPKI Object	Reference

.tal	Signed TAL	[this document]

9. Security Considerations

TBD

10. Acknowledgements

TBD

11. References

11.1. 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>>.

- [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <<https://www.rfc-editor.org/info/rfc4648>>.
- [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>>.
- [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>>.
- [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>>.
- [RFC7730] Huston, G., Weiler, S., Michaelson, G., and S. Kent, "Resource Public Key Infrastructure (RPKI) Trust Anchor Locator", RFC 7730, DOI 10.17487/RFC7730, January 2016, <<https://www.rfc-editor.org/info/rfc7730>>.
- [X.509] ITU-T Recommendation X.509 (2000), "Recommendation X.509: The Directory - Authentication Framework", 2000.
- [X.690] ITU-T Recommendation X.690 (2002) | ISO/IEC 8825-1:2002, "Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", 2002.

11.2. Informative References

- [RFC5011] StJohns, M., "Automated Updates of DNS Security (DNSSEC) Trust Anchors", STD 74, RFC 5011, DOI 10.17487/RFC5011, September 2007, <<https://www.rfc-editor.org/info/rfc5011>>.

[RFC6489] Huston, G., Michaelson, G., and S. Kent, "Certification Authority (CA) Key Rollover in the Resource Public Key Infrastructure (RPKI)", BCP 174, RFC 6489, DOI 10.17487/RFC6489, February 2012, <<https://www.rfc-editor.org/info/rfc6489>>.

Authors' Addresses

Tim Bruijnzeels
RIPE NCC

Email: tim@ripe.net

Carlos Martinez
LACNIC

Email: carlos@lacnic.net

Network Working Group
Internet-Draft
Updates: 7730 (if approved)
Intended status: Standards Track
Expires: May 20, 2018

T. Bruijnzeels
RIPE NCC
G. Michaelson
APNIC
November 16, 2017

Resource Public Key Infrastructure (RPKI) Trust Anchor Locator
draft-tbruijnzeels-sidrops-https-tal-00

Abstract

This document defines a Trust Anchor Locator (TAL) for the Resource Public Key Infrastructure (RPKI). This document obsoletes RFC 7730 by adding support for HTTPS URIs in a TAL.

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 May 20, 2018.

Copyright Notice

Copyright (c) 2017 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Terminology	2
2. Trust Anchor Locator	2
2.1. Trust Anchor Locator Format	2
2.2. TAL and Trust Anchor Certificate Considerations	3
2.3. Example	5
3. Relying Party Use	5
4. HTTPS Considerations	6
5. Security Considerations	7
6. Acknowledgements	7
7. References	8
7.1. Normative References	8
7.2. Informative References	9
Authors' Addresses	9

1. Introduction

This document defines a Trust Anchor Locator (TAL) for the Resource Public Key Infrastructure (RPKI) [RFC6480]. This format may be used to distribute trust anchor material using a mix of out-of-band and online means. Procedures used by Relying Parties (RPs) to verify RPKI signed objects SHOULD support this format to facilitate interoperability between creators of trust anchor material and RPs. This document obsoletes [RFC7730] by adding support for HTTPS URIs in a TAL.

1.1. Terminology

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be interpreted as described in [RFC2119].

2. Trust Anchor Locator

2.1. Trust Anchor Locator Format

This document does not propose a new format for trust anchor material. A trust anchor in the RPKI is represented by a self-signed X.509 Certification Authority (CA) certificate, a format commonly used in PKIs and widely supported by RP software. This document specifies a format for data used to retrieve and verify the authenticity of a trust anchor in a very simple fashion. That data is referred to as the TAL.

The motivation for defining the TAL is to enable selected data in the trust anchor to change, without needing to effect redistribution of

the trust anchor per se. In the RPKI, certificates contain extensions that represent Internet Number Resources (INRs) [RFC3779]. The set of INRs associated with an entity acting as a trust anchor is likely to change over time. Thus, if one were to use the common PKI convention of distributing a trust anchor to RPs in a secure fashion, then this procedure would need to be repeated whenever the INR set for the entity acting as a trust anchor changed. By distributing the TAL (in a secure fashion), instead of distributing the trust anchor, this problem is avoided, i.e., the TAL is constant so long as the trust anchor's public key and its location do not change.

The TAL is analogous to the TrustAnchorInfo data structure specified in [RFC5914], which is on the Standards Track. That specification could be used to represent the TAL, if one defined an rsync or HTTPS URI extension for that data structure. However, the TAL format was adopted by RPKI implementors prior to the PKIX trust anchor work, and the RPKI implementer community has elected to utilize the TAL format, rather than define the requisite extension. The community also prefers the simplicity of the ASCII encoding of the TAL, versus the binary (ASN.1) encoding for TrustAnchorInfo.

The TAL is an ordered sequence of:

1. a URI section,
2. a "<CRLF>" or "<LF>" line break,
3. a subjectPublicKeyInfo [RFC5280] in DER format [X.509], encoded in Base64 (see Section 4 of [RFC4648]). To avoid long lines, "<CRLF>" or "<LF>" line breaks MAY be inserted into the Base64-encoded string.

where the URI section is comprised of one of more of the ordered sequence of:

- 1.1. an rsync URI [RFC5781], or an HTTPS URI [RFC7230]
- 1.2. a "<CRLF>" or "<LF>" line break.
- 2.2. TAL and Trust Anchor Certificate Considerations

Each URI in the TAL MUST reference a single object. It MUST NOT reference a directory or any other form of collection of objects.

The referenced object MUST be a self-signed CA certificate that conforms to the RPKI certificate profile [RFC6487]. This certificate is the trust anchor in certification path discovery [RFC4158] and validation [RFC5280] [RFC3779].

The validity interval of this trust anchor SHOULD reflect the anticipated period of stability of the particular set of INRs that are associated with the putative trust anchor.

The INR extension(s) of this trust anchor MUST contain a non-empty set of number resources. It MUST NOT use the "inherit" form of the INR extension(s). The INR set described in this certificate is the set of number resources for which the issuing entity is offering itself as a putative trust anchor in the RPKI [RFC6480].

The public key used to verify the trust anchor MUST be the same as the subjectPublicKeyInfo in the CA certificate and in the TAL.

The trust anchor MUST contain a stable key. This key MUST NOT change when the certificate is reissued due to changes in the INR extension(s), when the certificate is renewed prior to expiration, or for any reason other than a key change.

Because the public key in the TAL and the trust anchor MUST be stable, this motivates operation of that CA in an offline mode. Thus, the entity that issues the trust anchor SHOULD issue a subordinate CA certificate that contains the same INRs (via the use of the "inherit" option in the INR extensions of the subordinate certificate). This allows the entity that issues the trust anchor to keep the corresponding private key of this certificate offline, while issuing all relevant child certificates under the immediate subordinate CA. This measure also allows the Certificate Revocation List (CRL) issued by that entity to be used to revoke the subordinate CA certificate in the event of suspected key compromise of this online operational key pair that is potentially more vulnerable.

The trust anchor MUST be published at a stable URI. When the trust anchor is reissued for any reason, the replacement CA certificate MUST be accessible using the same URI.

Because the trust anchor is a self-signed certificate, there is no corresponding CRL that can be used to revoke it, nor is there a manifest [RFC6486] that lists this certificate.

If an entity wishes to withdraw a self-signed CA certificate as a putative trust anchor, for any reason, including key rollover, the entity MUST remove the object from the location referenced in the TAL.

Where the TAL contains two or more URIs, then the same self-signed CA certificate MUST be found at each referenced location. In order to increase operational resilience, it is RECOMMENDED that the domain name parts of each of these URIs resolve to distinct IP addresses

that are used by a diverse set of repository publication points, and these IP addresses be included in distinct Route Origin Authorizations (ROAs) objects signed by different CAs.

2.3. Example

```
rsync://rpki.example.org/rpki/hedgehog/root.cer
<https://rpki.example.org/rpki/hedgehog/root.cer>
```

```
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAAovWQL2lh6knDx
GUG5hbtCXvvh4AOzjhDkSHlj22gn/loiM9IeDATIwP44vhQ6L/xvuk7W6
Kfa5ygmqQ+xOZOwTWPCrUbqaQyPNxokuivzyvqVZVDecOEqs78q58mSp9
nbtxmLRW7B67SJCBSzfa5XpVyXYEgYAJkk3fpmefU+AcctxvvHB5OVPIa
BfPcs80ICMgHQX+fphvute9XLxjfkJKJWkhZqZ0v7pZm2uhkcPx1PMGcrG
ee0WSDC3fr3erLueagpiLsFjwwpX6F+Ms8vqz45H+DKmYKvPSstZjCCq9
aJ0qANT90tnfSDOS+aLRPjZryCNyvvBHxZXqj5YCGKtwIDAQAB
```

3. Relying Party Use

In order to use the TAL to retrieve and validate a (putative) trust anchor, an RP SHOULD:

1. Retrieve the object referenced by (one of) the 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 trust anchor. These tests apply to the validity of attestations made in the context of the RPKI relating to all resources described in the INR extension of this certificate.

An RP SHOULD perform these functions for each instance of TAL that it is holding for this purpose every time the RP performs a resynchronization across the local repository cache. In any case, an RP also SHOULD perform these functions prior to the expiration of the locally cached copy of the retrieved trust anchor referenced by the TAL.

In the case where a TAL contains multiple URIs, an RP MAY use a locally defined preference rule to select the URI to retrieve the

self-signed RPKI CA certificate that is to be used as a trust anchor. Some examples are:

- o Using the order provided in the TAL
- o Selecting the URI randomly from the available list
- o Creating a prioritized list of URIs based on RP-specific parameters, such as connection establishment delay

If the connection to the preferred URI fails, or the retrieved CA certificate public key does not match the TAL public key, the RP SHOULD retrieve the CA certificate from the next URI, according to the local preference ranking of URIs.

4. HTTPS Considerations

REMOVE LATER: The following text is inspired by the equivalent section in [RFC8182], but adapted for this case.

Note that a Man in the Middle (MITM) cannot produce a CA certificate that would be considered valid according to the process described in Section 3. However, a MITM can perform withhold or replay attacks targeting a Relying Party and keep the Relying Party from learning about an update CA certificate. Because of this, Relying Parties SHOULD do TLS certificate and host name validation when they fetch a CA certificate using an HTTPS URI on a TAL.

Relying Party tools SHOULD log any TLS certificate or host name validation issues found, so that an operator can investigate the cause. However, such validation issues are often due to configuration errors or a lack of a common TLS trust anchor. In these cases, it is better if the Relying Party retrieves the CA certificate regardless and performs validation on it. Therefore, the Relying Party MUST continue to retrieve the data in case of errors.

It is RECOMMENDED that Relying Parties and Repository Servers follow the Best Current Practices outlined in [RFC7525] on the use of HTTP over TLS (HTTPS) [RFC7230]. Relying Parties SHOULD do TLS certificate and host name validation using subjectAltName dNSName identities as described in [RFC6125]. The rules and guidelines defined in [RFC6125] apply here, with the following considerations:

- o Relying Parties and Repository Servers SHOULD support the DNS-ID identifier type. The DNS-ID identifier type SHOULD be present in Repository Server certificates.

- o DNS names in Repository Server certificates SHOULD NOT contain the wildcard character "*".
- o A Common Name (CN) field may be present in a Repository Server certificate's subject name but SHOULD NOT be used for authentication within the rules described in [RFC6125].
- o This protocol does not require the use of SRV-IDs.
- o This protocol does not require the use of URI-IDs.

Note, however, that this validation is done on a best-effort basis and serves to highlight potential issues, but CA certificate validation in relation to a TAL as described in Section 3 does not depend on this. Therefore, Relying Parties MAY deviate from the validation steps listed above.

5. Security Considerations

Compromise of a trust anchor private key permits unauthorized parties to masquerade as a trust anchor, with potentially severe consequences. Reliance on an inappropriate or incorrect trust anchor has similar potentially severe consequences.

This TAL does not directly provide a list of resources covered by the referenced self-signed CA certificate. Instead, the RP is referred to the trust anchor itself and the INR extension(s) within this certificate. This provides necessary operational flexibility, but it also allows the certificate issuer to claim to be authoritative for any resource. Relying parties should either have great confidence in the issuers of such certificates that they are configuring as trust anchors, or they should issue their own self-signed certificate as a trust anchor and, in doing so, impose constraints on the subordinate certificates.

6. Acknowledgements

This approach to trust anchor material was originally described by Robert Kisteleki.

The authors acknowledge the contributions of Rob Austein and Randy Bush, who assisted with drafting this document and with helpful review comments.

The authors acknowledge with work of Roque Gagliano, Terry Manderson, and Carlos Martinez Cagnazzo in developing the ideas behind the inclusion of multiple URIs in the TAL.

7. References

7.1. 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>>.
- [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <<https://www.rfc-editor.org/info/rfc4648>>.
- [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>>.
- [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>>.
- [RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <<https://www.rfc-editor.org/info/rfc6125>>.
- [RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, DOI 10.17487/RFC6480, February 2012, <<https://www.rfc-editor.org/info/rfc6480>>.
- [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>>.
- [RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, DOI 10.17487/RFC7230, June 2014, <<https://www.rfc-editor.org/info/rfc7230>>.

- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC7730] Huston, G., Weiler, S., Michaelson, G., and S. Kent, "Resource Public Key Infrastructure (RPKI) Trust Anchor Locator", RFC 7730, DOI 10.17487/RFC7730, January 2016, <<https://www.rfc-editor.org/info/rfc7730>>.
- [X.509] TU-T Recommendation X.509, "The Directory: Public-key and attribute certificate frameworks", October 2012.

7.2. Informative References

- [RFC4158] Cooper, M., Dzambasow, Y., Hesse, P., Joseph, S., and R. Nicholas, "Internet X.509 Public Key Infrastructure: Certification Path Building", RFC 4158, DOI 10.17487/RFC4158, September 2005, <<https://www.rfc-editor.org/info/rfc4158>>.
- [RFC5914] Housley, R., Ashmore, S., and C. Wallace, "Trust Anchor Format", RFC 5914, DOI 10.17487/RFC5914, June 2010, <<https://www.rfc-editor.org/info/rfc5914>>.
- [RFC6486] Austein, R., Huston, G., Kent, S., and M. Lepinski, "Manifests for the Resource Public Key Infrastructure (RPKI)", RFC 6486, DOI 10.17487/RFC6486, February 2012, <<https://www.rfc-editor.org/info/rfc6486>>.
- [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>>.

Authors' Addresses

Tim Bruijnzeels
RIPE NCC

Email: tim@ripe.net

George Michaelson
APNIC

Email: ggm@apnic.net