

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: July 10, 2019

A. Azimov
E. Uskov
Qrator Labs
R. Bush
Internet Initiative Japan
K. Patel
Arrcus
J. Snijders
NTT
R. Housley
Vigil Security
January 6, 2019

A Profile for Autonomous System Provider Authorization
draft-azimov-sidrops-aspa-profile-01

Abstract

This document defines a standard profile for Autonomous System Provider Authorization in the Resource Public Key Infrastructure. An Autonomous System Provider Authorization is a digitally signed object that provides a means of verifying that a Customer Autonomous System holder has authorized a Provider Autonomous System to be its upstream provider and for the Provider to send prefixes received from the Customer Autonomous System in all directions including providers and peers.

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.

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 July 10, 2019.

Copyright Notice

Copyright (c) 2019 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. The ASPA Content Type	3
3. The ASPA eContent	3
3.1. version	4
3.2. AFI	4
3.3. customerASID	4
3.4. providerASID	4
4. ASPA Validation	5
5. ASN.1 Module for the ASPA Content Type	5
6. IANA Considerations	6
7. Security Considerations	7
8. Acknowledgments	7
9. References	7
9.1. Normative References	7
9.2. Informative References	8
Authors' Addresses	8

1. Introduction

The primary purpose of the Resource Public Key Infrastructure (RPKI) is to improve routing security. (See [RFC6480] for more information.) As part of this infrastructure, a mechanism is needed to verify that a Provider AS (PAS) has permission from a Customer AS (CAS) holder to send routes in all directions. The digitally signed Autonomous System Provider Authorization (ASPA) object provides this verification mechanism.

The ASPA uses the template for RPKI digitally signed objects [RFC6488], which defines a Cryptographic Message Syntax (CMS) [RFC5652] wrapper for the ASPA content as well as a generic validation procedure for RPKI signed objects. As ASPAs need to be validated with RPKI certificates issued by the current infrastructure, we assume the mandatory-to-implement algorithms in [RFC6485], or its successor.

To complete the specification of the ASPA (see Section 4 of [RFC6488]), this document defines:

1. The object identifier (OID) that identifies the ASPA signed object. This OID appears in the eContentType field of the encapContentInfo object as well as the content-type signed attribute within the signerInfo structure).
2. The ASN.1 syntax for the ASPA content, which is the payload signed by the CAS. The ASPA content is encoded using the ASN.1 [X680] Distinguished Encoding Rules (DER) [X690].
3. The steps required to validate an ASPA beyond the validation steps specified in [RFC6488]).

2. The ASPA Content Type

The content-type for an ASPA is defined as id-cct-ASPA, which has the numerical value of 1.2.840.113549.1.9.16.1.TBD. This OID MUST appear both within the eContentType in the encapContentInfo structure as well as the content-type signed attribute within the signerInfo structure (see [RFC6488]).

3. The ASPA eContent

The content of an ASPA identifies the Customer AS (CAS) as well as the Provider AS (PAS) that is authorized to further propagate announcements received from the customer. If customer has multiple providers, it issues multiple ASPAs, one for each provider AS. An ASPA is formally defined as:

```
ct-ASPA CONTENT-TYPE ::=
    { ASProviderAttestation IDENTIFIED BY id-ct-ASPA }

id-ct-ASPA OBJECT IDENTIFIER ::= { id-ct TBD }

ASProviderAttestation ::= SEQUENCE {
    version [0] ASPAVersion DEFAULT v0,
    AFI AddressFamilyIdentifier,
    customerASID ASID,
    providerASID ASID }

ASPAVersion ::= INTEGER { v0(0) }

AddressFamilyIdentifier ::= INTEGER

ASID ::= INTEGER
```

Note that this content appears as the eContent within the encapContentInfo as specified in [RFC6488].

3.1. version

The version number of the ASProviderAttestation MUST be v0.

3.2. AFI

The AFI field contains Address Family Identifier for which the relation between customer and provider ASes is authorized. Presently defined values for the Address Family Identifier field are specified in the IANA's Address Family Numbers registry [IANA-AF].

3.3. customerASID

The customerASID field contains the AS number of the Autonomous System that authorizes an upstream provider (listed in the providerASID) to propagate prefixes in the specified address family other ASes.

3.4. providerASID

The providerASID contains the AS number that is authorized to further propagate announcements in the specified address family received from the customer.

4. ASPA Validation

Before a relying party can use an ASPA to validate a routing announcement, the relying party MUST first validate the ASPA object itself. To validate an ASPA, the relying party MUST perform all the validation checks specified in [RFC6488] as well as the following additional ASPA-specific validation step.

- o The autonomous system identifier delegation extension [RFC3779] is present in the end-entity (EE) certificate (contained within the ASPA), and the customer AS number in the ASPA is contained within the set of AS numbers specified by the EE certificate's autonomous system identifier delegation extension.

5. ASN.1 Module for the ASPA Content Type

```
RPKI-ASPA-2018
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
    pkcs-9(9) smime(16) modules(0) id-mod-rpki-aspa-2018(TBD2) }
DEFINITIONS IMPLICIT TAGS ::=
BEGIN
IMPORTS

CONTENT-TYPE
FROM CryptographicMessageSyntax-2010 -- RFC 6268
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
    pkcs-9(9) smime(16) modules(0) id-mod-cms-2009(58) } ;

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

--
-- ASPA Content Type
--

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

id-ct OBJECT IDENTIFIER ::= { id-smime 1 }

id-ct-ASPA OBJECT IDENTIFIER ::= { id-ct TBD }

ct-ASPA CONTENT-TYPE ::=
  { TYPE ASPProviderAttestation IDENTIFIED BY id-ct-ASPA }

ASPProviderAttestation ::= SEQUENCE {
  version [0] ASPAVersion DEFAULT v0,
  AFI AddressFamilyIdentifier,
  customerASID ASID,
  providerASID ASID }

ASPAVersion ::= INTEGER { v0(0) }

AddressFamilyIdentifier ::= INTEGER

ASID ::= INTEGER

END
```

6. IANA Considerations

Please add the id-mod-rpki-aspa-2018 to the SMI Security for S/MIME Module Identifier (1.2.840.113549.1.9.16.0) registry (<https://www.iana.org/assignments/smi-numbers/smi-numbers.xml#security-smime-0>) as follows:

Decimal	Description	Specification
TBD2	id-mod-rpki-aspa-2018	[ThisRFC]

Please add the ASPA to the SMI Security for S/MIME CMS Content Type (1.2.840.113549.1.9.16.1) registry (<https://www.iana.org/assignments/smi-numbers/smi-numbers.xml#security-smime-1>) as follows:

Decimal	Description	Specification
TBD	id-ct-ASPA	[ThisRFC]

Please add the ASPA to the RPKI Signed Object registry (<https://www.iana.org/assignments/rpki/rpki.xhtml#signed-objects>) as follows:

Name	OID	Specification
ASPA	1.2.840.113549.1.9.16.1.TBD	[ThisRFC]

7. Security Considerations

8. Acknowledgments

9. References

9.1. Normative References

- [IANA-AF] IANA, "Address Family Numbers",
<<http://www.iana.org/numbers.html>>.
- [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>>.
- [RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <<https://www.rfc-editor.org/info/rfc5652>>.

- [RFC6485] Huston, G., "The Profile for Algorithms and Key Sizes for Use in the Resource Public Key Infrastructure (RPKI)", RFC 6485, DOI 10.17487/RFC6485, February 2012, <<https://www.rfc-editor.org/info/rfc6485>>.
- [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>>.
- [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>>.
- [X680] ITU-T, "Information technology -- Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, 2015.
- [X690] 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, 2015.

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>>.

Authors' Addresses

Alexander Azimov
Qrator Labs

Email: a.e.azimov@gmail.com

Eugene Uskov
Qrator Labs

Email: eu@qrator.net

Randy Bush
Internet Initiative Japan

Email: randy@psg.com

Keyur Patel
Arrcus, Inc.

Email: keyur@arrcus.com

Job Snijders
NTT Communications
Theodorus Majofskistraat 100
Amsterdam 1065 SZ
The Netherlands

Email: job@ntt.net

Russ Housley
Vigil Security, LLC
918 Spring Knoll Drive
Herndon, VA 20170
USA

Email: housley@vigilsec.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 25, 2019

A. Azimov
E. Bogomazov
Qrator Labs
R. Bush
Internet Initiative Japan
K. Patel
Arccus, Inc.
J. Snijders
NTT
October 22, 2018

Verification of AS_PATH Using the Resource Certificate Public Key
Infrastructure and Autonomous System Provider Authorization
draft-azimov-sidrops-aspa-verification-01

Abstract

This document defines the semantics of an Autonomous System Provider Authorization object in the Resource Public Key Infrastructure to verify the AS_PATH attribute of routes advertised in the Border Gateway Protocol.

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.

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 April 25, 2019.

Copyright Notice

Copyright (c) 2018 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. Anomaly Propagation	3
3. Autonomous System Provider Authorization	4
4. Customer-Provider Verification Procedure	4
5. AS_PATH Verification	5
6. Disavowal of Provider Authorizaion	6
7. Siblings (Complex Relations)	6
8. Security Considerations	7
9. Acknowledgments	7
10. References	7
10.1. Normative References	7
10.2. Informative References	7
Authors' Addresses	9

1. Introduction

The Border Gateway Protocol (BGP) was designed with no mechanisms to validate BGP attributes. Two consequences are BGP Hijacks and BGP Route Leaks [RFC7908]. BGP extensions are able to partially solve these problems. For example, ROA-based Origin Validation [RFC6483] can be used to detect and filter accidental mis-originations, and [I-D.ymbk-idr-bgp-eotr-policy] can be used to detect accidental route leaks. While these upgrades to BGP are quite useful, they still rely on transitive BGP attributes, i.e. AS_PATH, that can be manipulated by attackers.

BGPsec [RFC8205] was designed to solve the problem of AS_PATH validation. Unfortunately, strict cryptographic validation brought unaffordable computational overhead for BGP routers. BGPsec also proved to be vulnerable to downgrade attacks that can nullify all the work of AS_PATH signing. As a result, to abuse the AS_PATH or any

other signed transit attribute, an attacker merely needs to downgrade to 'old' BGP-4.

An alternative approach was introduced with soBGP [I-D.white-sobgp-architecture]. Instead of strong cryptographic AS_PATH validation, it was suggested to create an AS_PATH security function based on a shared database of ASN adjacencies. While such an approach has reasonable computational cost, the two side adjacencies don't provide a way to automate anomaly detection without high adoption rate - an attacker can easily up a one-way adjacency. SO-BGP suggested sharing data about adjacencies using additional BGP messages, which is recursively complex thus significantly increasing adoption complexity. In addition, the general goal to verify all AS_PATHs was not achievable given the indirect adjacencies at internet exchange points.

Instead of the general goal of checking AS_PATH correctness, this document focuses on solving real-world operational problems - automatic detection of malicious hijacks and route leaks. To achieve this goal a new AS_PATH verification procedure is defined which is able to automatically detect invalid (malformed) AS_PATHs in announcements that are received from customers and peers. This procedure uses a shared signed database of customer-to-provider relationships that is built using a new RPKI object - Autonomous System Provider Authorization (ASPA). This technique provides benefits for the participants even in a state of early adoption.

2. Anomaly Propagation

Both route leaks and hijacks have similar effects on ISP operations - they redirect traffic, resulting in increased latency, packet loss, or possible MiTM attacks. But the level of risk depends significantly on the propagation of these BGP anomalies. For example, a hijack that is propagated only to customers may concentrate traffic in a particular ISP's customer cone; while if the anomaly is propagated through peers, upstreams, or reaches Tier-1 networks, thus distributing globally, traffic may be redirected at the level of entire countries and/or global providers.

The ability to constrain propagation of BGP anomalies to upstreams and peers, without requiring support from the source of the anomaly (which is critical if source has malicious intent), should significantly improve the security of inter-domain routing and solve the majority of problems.

3. Autonomous System Provider Authorization

As described in [RFC6480], the RPKI is based on a hierarchy of resource certificates that are aligned to the Internet Number Resource allocation structure. Resource certificates are X.509 certificates that conform to the PKIX profile [RFC5280], and to the extensions for IP addresses and AS identifiers [RFC3779]. A resource certificate is a binding by an issuer of IP address blocks and Autonomous System (AS) numbers to the subject of a certificate, identified by the unique association of the subject's private key with the public key contained in the resource certificate. The RPKI is structured so that each current resource certificate matches a current resource allocation or assignment.

ASPAs are digitally signed objects that bind a selected AFI Provider AS number to a Customer AS number (in terms of BGP announcements not business), and are signed by the holder of the Customer AS. An ASPA attests that a Customer AS holder (CAS) has authorized a particular Provider AS (PAS) to propagate the Customer's IPv4/IPv6 announcements onward, e.g. to the Provider's upstream providers or peers. The ASPA record profile is described in [I-D.azimov-sidrops-asma-profile].

4. Customer-Provider Verification Procedure

This section describes an abstract procedure that checks that pair of ASNs (AS1, AS2) is included in the set of signed ASPAs. The semantics of its use are defined in next section. The procedure takes (AS1, AS2, ROUTE_AFI) as input parameters and returns three types of results: "valid", "invalid" and "unknown".

A relying party (RP) must have access to a local cache of the complete set of cryptographically valid ASPAs when performing customer-provider verification procedure.

1. Retrieve all cryptographically valid ASPAs in a selected AFI with a customer value of AS1. This selection forms the set of "candidate ASPAs."
2. If the set of candidate ASPAs is empty, then the procedure exits with an outcome of "unknown."
3. If there is at least one candidate ASPA where the provider field is AS2, then the procedure exits with an outcome of "valid."
4. Otherwise, the procedure exits with an outcome of "invalid."

Since an AS1 may have different set providers in different AFI, it should also have different set of corresponding ASPAs. In this case,

the output of this procedure with input (AS1, AS2, ROUTE_AFI) may have different output for different ROUTE_AFI values.

5. AS_PATH Verification

The AS_PATH attribute identifies the autonomous systems through which an UPDATE message has passed. AS_PATH may contain two types of components: ordered AS_SEQs and unordered AS_SETs, as defined in [RFC4271].

The value of each AS_SEQ component can be described as set of pairs {(AS(I), prepend(I)), (AS(I-1), prepend(I-1))...}. In this case, the sequence {AS(I), AS(I-1),...} represents different ASNs, that packet should pass towards the destination. When a route is received from a customer or a literal peer, each pair (AS(I-1), AS(I)) MUST belong to customer-provider or sibling relationship. If there are other types of relationships, it means that the route was leaked or the AS_PATH attribute was malformed. The goal of the above procedure is to check the correctness of this statement.

For 32-bit AS number compatible BGP speakers, if a route from ROUTE_AFI address family is received from a customer or peer, its AS_PATH MUST be verified as follows:

1. If the closest AS in the AS_PATH is not the receiver's neighbor ASN then procedure halts with the outcome "invalid";
2. If in one of AS_SEQ segments there is a pair (AS(I-1), AS(I)), and customer-provider verification procedure (Section 4) with parameters (AS(I-1), AS(I), ROUTE_AFI) returns "invalid" then the procedure also halts with the outcome "invalid";
3. If the AS_PATH has at least one AS_SET segment then procedure halts with the outcome "unverifiable";
4. Otherwise, the procedure halts with an outcome of "valid".

For BGP speakers that are not 32-bit AS compatible, the above procedure is slightly different. In point 2 if at least one AS(I-1), AS(I) is equal to AS_TRANS(23456), the corresponding pair must be passed without check using the customer-provider verification procedure.

If the output of the AS_PATH verification procedure is "invalid" the LOCAL_PREF SHOULD be set to 0 or the route MAY be dropped. If an "invalid" route has no alternative route(s) and it is propagated to other ASes despite the above, it MUST be marked with the GRACEFUL_SHUTDOWN community to avoid possible stable oscillations,

when an unchecked route received from a provider becomes preferred over an invalid route received from a customer. This also allows customers to detect malformed routes received from upstream providers.

If the output of the AS_PATH verification procedure is 'unverifiable' it means that AS_PATH can't be fully verified. Such routes should be treated with caution and SHOULD be processed the same way as "invalid" routes. This policy goes with full correspondence to [I-D.kumari-deprecate-as-set-confed-set].

The above AS_PATH verification procedure is able to check routes received from customers and peers. The ASPA mechanism combined with BGP Roles [I-D.ietf-idr-bgp-open-policy] and ROA-based Origin Validation [RFC6483] provide a fully automated solution to detect and filter hijacks and route leaks, including malicious ones.

6. Disavowal of Provider Authorizaion

An ASPA is a positive attestation that an ASholder has authorized its provider to redistribute received routes to the provider's providers and peers. This does not preclude the provider AS from redistribution to its other customers. By creating an ASPA where the provider AS is 0, the customer indicates that no provider should further announce its routes. Specifically, AS 0 is reserved to identify provider-free networks, Internet exchange meshes, etc.

An ASPA with a provider AS of 0 is a statement by the customer AS that the its routes should not be received by any relying party AS from any of its customers or peers.

By convention, an ASPA with a provider AS of 0 should be the only ASPA issued by a given AS holder; although this is not a strict requirement. A provider 0 ASPA may coexist with ASPAs that have different provider AS values; though in such cases, the presence or absence of the provider AS 0 ASPA does not alter the AS_PATH verification procedure.

7. Siblings (Complex Relations)

There are peering relationships which can not be described as strictly simple peer-peer or customer-provider; e.g. when both parties are intentionally sending prefixes received from each other to their peers and/or upstreams.

In this case, two symmetric ASPAs records {(AS1, AS2), (AS2, AS1)} must be created by AS1 and AS2 respectively.

8. Security Considerations

ASPA issuers should be aware of the verification implication in issuing an ASPA – an ASPA implicitly invalidates all routes passed to upstream providers other than the provider ASs listed in the collection of ASPAs. It is the Customer AS's duty to maintain a correct set of ASPAs.

While the ASPA provides a check of an AS_PATH for routes received from customers and peers, it doesn't provide full support for routes that are received from upstream providers. So, this mechanism guarantees detection of both malicious and accidental route leaks and provides partial support for detection of malicious hijacks: upstream transit ISPs will still be able to send hijacked prefixes with malformed AS_PATHs to their customers.

9. Acknowledgments

The authors wish to thank authors of [RFC6483] since its text was used as an example while writing this document.

10. References

10.1. 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>>.
- [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>>.

10.2. Informative References

- [I-D.azimov-sidrops-aspa-profile] Azimov, A., Uskov, E., Bush, R., Patel, K., Snijders, J., and R. Housley, "A Profile for Autonomous System Provider Authorization", draft-azimov-sidrops-aspa-profile-00 (work in progress), June 2018.
- [I-D.ietf-idr-bgp-open-policy] Azimov, A., Bogomazov, E., Bush, R., Patel, K., and K. Sriram, "Route Leak Prevention using Roles in Update and Open messages", draft-ietf-idr-bgp-open-policy-02 (work in progress), January 2018.

- [I-D.kumari-deprecate-as-set-confed-set]
Kumari, W. and K. Sriram, "Deprecation of AS_SET and AS_CONFED_SET in BGP", draft-kumari-deprecate-as-set-confed-set-12 (work in progress), July 2018.
- [I-D.white-sobgp-architecture]
White, R., "Architecture and Deployment Considerations for Secure Origin BGP (soBGP)", draft-white-sobgp-architecture-02 (work in progress), June 2006.
- [I-D.ymbk-idr-bgp-eotr-policy]
Azimov, A., Bogomazov, E., Bush, R., and K. Patel, "Route Leak Detection and Filtering using Roles in Update and Open messages", draft-ymbk-idr-bgp-eotr-policy-02 (work in progress), March 2018.
- [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>>.
- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <<https://www.rfc-editor.org/info/rfc4271>>.
- [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>>.
- [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>>.
- [RFC6483] Huston, G. and G. Michaelson, "Validation of Route Origination Using the Resource Certificate Public Key Infrastructure (PKI) and Route Origin Authorizations (ROAs)", RFC 6483, DOI 10.17487/RFC6483, February 2012, <<https://www.rfc-editor.org/info/rfc6483>>.
- [RFC7908] Sriram, K., Montgomery, D., McPherson, D., Osterweil, E., and B. Dickson, "Problem Definition and Classification of BGP Route Leaks", RFC 7908, DOI 10.17487/RFC7908, June 2016, <<https://www.rfc-editor.org/info/rfc7908>>.

[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>>.

Authors' Addresses

Alexander Azimov
Qrator Labs

Email: aa@qrator.net

Eugene Bogomazov
Qrator Labs

Email: eb@qrator.net

Randy Bush
Internet Initiative Japan

Email: randy@psg.com

Keyur Patel
Arrcus, Inc.

Email: keyur@arrcus.com

Job Snijders
NTT Communications
Theodorus Majofskistraat 100
Amsterdam 1065 SZ
The Netherlands

Email: job@ntt.net

Internet Engineering Task Force (IETF)
Internet-Draft
Intended status: Best Current Practice
Expires: 4 November 2022

Y. Gilad
Hebrew University of Jerusalem
S. Goldberg
Boston University
K. Sriram
USA NIST
J. Snijders
Fastly
B. Maddison
Workonline Communications
3 May 2022

The Use of maxLength in the RPKI
draft-ietf-sidrops-rpkimaxlen-10

Abstract

This document recommends ways to reduce the forged-origin hijack attack surface by prudently limiting the set of IP prefixes that are included in a Route Origin Authorization (ROA). One recommendation is to avoid using the maxLength attribute in ROAs except in some specific cases. The recommendations complement and extend those in RFC 7115. The document also discusses the creation of ROAs for facilitating the use of Distributed Denial of Service (DDoS) mitigation services. Considerations related to ROAs and origin validation in the context of destination-based Remote Triggered Black Hole (RTBH) filtering are also highlighted.

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 4 November 2022.

Copyright Notice

Copyright (c) 2022 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	4
1.2. Documentation Prefixes	4
2. Suggested Reading	4
3. Forged-Origin Subprefix Hijack	4
4. Measurements of the RPKI	6
5. Recommendations about Minimal ROAs and maxLength	7
5.1. Facilitating Ad-hoc Routing Changes and DDoS Mitigation	7
5.2. Defensive de-aggregation in response to prefix hijacks	10
6. Considerations for RTBH Filtering Scenarios	11
7. IANA Considerations	11
8. Security Considerations	11
9. Acknowledgments	12
10. References	12
10.1. Normative References	12
10.2. Informative References	12
Authors' Addresses	13

1. Introduction

The RPKI [RFC6480] uses Route Origin Authorizations (ROAs) to create a cryptographically verifiable mapping from an IP prefix to a set of autonomous systems (ASes) that are authorized to originate that prefix. Each ROA contains a set of IP prefixes, and an AS number of an AS authorized to originate all the IP prefixes in the set [RFC6482]. The ROA is cryptographically signed by the party that holds a certificate for the set of IP prefixes.

The ROA format also supports a maxLength attribute. According to [RFC6482], "When present, the maxLength specifies the maximum length of the IP address prefix that the AS is authorized to advertise."

Thus, rather than requiring the ROA to list each prefix that the AS is authorized to originate, the maxLength attribute provides a shorthand that authorizes an AS to originate a set of IP prefixes.

However, measurements of RPKI deployments have found that the use of the maxLength in ROAs tends to lead to security problems. In particular, measurements taken in June 2017 showed that 84% of the prefixes specified in ROAs that use the maxLength attribute, were vulnerable to a forged-origin subprefix hijack [HARMFUL]. The forged-origin prefix or subprefix hijack involves inserting the legitimate AS as specified in the ROA as the origin AS in the AS_PATH, and can be launched against any IP prefix/subprefix that has a ROA. Consider a prefix/subprefix that has a ROA but is unused, i.e., not announced in BGP by a legitimate AS. A forged origin hijack involving such a prefix/subprefix can propagate widely throughout the Internet. On the other hand, if the prefix/subprefix were announced by the legitimate AS, then the propagation of the forged-origin hijack is somewhat limited because of its increased AS_PATH length relative to the legitimate announcement. Of course, forged-origin hijacks are harmful in both cases but the extent of harm is greater for unannounced prefixes.

For this reason, this document recommends that, whenever possible, operators SHOULD use "minimal ROAs" that authorize only those IP prefixes that are actually originated in BGP, and no other prefixes. Further, it recommends ways to reduce the forged-origin attack surface by prudently limiting the address space that is included in Route Origin Authorizations (ROAs). One recommendation is to avoid using the maxLength attribute in ROAs except in some specific cases. The recommendations complement and extend those in [RFC7115]. The document also discusses the creation of ROAs for facilitating the use of Distributed Denial of Service (DDoS) mitigation services. Considerations related to ROAs and origin validation in the context of destination-based Remote Triggered Black Hole (RTBH) filtering are also highlighted.

One ideal place to implement the ROA related recommendations is in the user interfaces for configuring ROAs. Thus, this document further recommends that designers and/or providers of such user interfaces SHOULD provide warnings to draw the user's attention to the risks of using the maxLength attribute.

Best current practices described in this document require no changes to the RPKI specification and will not increase the number of signed ROAs in the RPKI, because ROAs already support lists of IP prefixes [RFC6482].

1.1. Requirements

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].

1.2. Documentation Prefixes

The documentation prefixes recommended in [RFC5737] are insufficient for use as example prefixes in this document. Therefore, this document uses [RFC1918] address space for constructing example prefixes.

2. Suggested Reading

It is assumed that the reader understands BGP [RFC4271], RPKI [RFC6480], Route Origin Authorizations (ROAs) [RFC6482], RPKI-based Prefix Validation [RFC6811], and BGPsec [RFC8205].

3. Forged-Origin Subprefix Hijack

A detailed description and discussion of forged-origin subprefix hijacks are presented here, especially considering the case when the subprefix is not announced in BGP. The forged-origin subprefix hijack is relevant to a scenario in which:

- (1) the RPKI [RFC6480] is deployed, and
- (2) routers use RPKI origin validation to drop invalid routes [RFC6811], but
- (3) BGPsec [RFC8205] (or any similar method to validate the truthfulness of the BGP AS_PATH attribute) is not deployed.

Note that this set of assumptions accurately describes a substantial and growing number of large Internet networks at the time of writing.

The forged-origin subprefix hijack [RFC7115] [GCHSS] is described here using a running example.

Consider the IP prefix 192.168.0.0/16 which is allocated to an organization that also operates AS 64496. In BGP, AS 64496 originates the IP prefix 192.168.0.0/16 as well as its subprefix 192.168.225.0/24. Therefore, the RPKI should contain a ROA authorizing AS 64496 to originate these two IP prefixes.

Suppose, however, the organization issues and publishes a ROA including a maxLength value of 24:

ROA:(192.168.0.0/16-24, AS 64496)

We refer to the above as a "loose ROA" since it authorizes AS 64496 to originate any subprefix of 192.168.0.0/16 up to and including length /24, rather than only those prefixes that are intended to be announced in BGP.

Because AS 64496 only originates two prefixes in BGP: 192.168.0.0/16 and 192.168.225.0/24, all other prefixes authorized by the "loose ROA" (for instance, 192.168.0.0/24), are vulnerable to the following forged-origin subprefix hijack [RFC7115] [GCHSS]:

The hijacker AS 64511 sends a BGP announcement "192.168.0.0/24: AS 64511, AS 64496", falsely claiming that AS 64511 is a neighbor of AS 64496 and falsely claiming that AS 64496 originates the IP prefix 192.168.0.0/24. In fact, the IP prefix 192.168.0.0/24 is not originated by AS 64496.

The hijacker's BGP announcement is valid according to the RPKI since the ROA (192.168.0.0/16-24, AS 64496) authorizes AS 64496 to originate BGP routes for 192.168.0.0/24.

Because AS 64496 does not actually originate a route for 192.168.0.0/24, the hijacker's route is the **only** route to the 192.168.0.0/24. The longest-prefix-match routing ensures that the hijacker's route to the subprefix 192.168.0.0/24 is always preferred over the legitimate route to 192.168.0.0/16 originated by AS 64496.

Thus, the hijacker's route propagates through the Internet, the traffic destined for IP addresses in 192.168.0.0/24 will be delivered to the hijacker.

The forged-origin **subprefix** hijack would have failed if a "minimal ROA" described below was used instead of the "loose ROA". In this example, a "minimal ROA" would be:

ROA:(192.168.0.0/16, 192.168.225.0/24, AS 64496)

This ROA is "minimal" because it includes only those IP prefixes that AS 64496 originates in BGP, but no other IP prefixes [RFC6907].

The "minimal ROA" renders AS 64511's BGP announcement invalid, because:

(1) this ROA "covers" the attacker's announcement (since 192.168.0.0/24 is a subprefix of 192.168.0.0/16), and

(2) there is no ROA "matching" the attacker's announcement (there is no ROA for AS 64511 and IP prefix 192.168.0.0/24) [RFC6811].

If routers ignore invalid BGP announcements, the minimal ROA above ensures that the subprefix hijack will fail.

Thus, if a "minimal ROA" had been used, the attacker would be forced to launch a forged-origin *prefix* hijack in order to attract traffic, as follows:

The hijacker AS 64511 sends a BGP announcement "192.168.0.0/16: AS 64511, AS 64496", falsely claiming that AS 64511 is a neighbor of AS 64496.

This forged-origin *prefix* hijack is significantly less damaging than the forged-origin *subprefix* hijack:

AS 64496 legitimately originates 192.168.0.0/16 in BGP, so the hijacker AS 64511 is not presenting the *only* route to 192.168.0.0/16.

Moreover, the path originated by AS 64511 is one hop longer than the path originated by the legitimate origin AS 64496.

As discussed in [LSG16], this means that the hijacker will attract less traffic than he would have in the forged-origin *subprefix* hijack, where the hijacker presents the *only* route to the hijacked subprefix.

In summary, a forged-origin subprefix hijack has the same impact as a regular subprefix hijack, despite the increased AS_PATH length of the illegitimate route. A forged-origin *subprefix* hijack is also more damaging than the forged-origin *prefix* hijack.

4. Measurements of the RPKI

Network measurements taken in June 2017 showed that 12% of the IP prefixes authorized in ROAs have a maxLength longer than their prefix length. Of these, the vast majority (84%) were non-minimal, as they included subprefixes that are not announced in BGP by the legitimate AS, and were thus vulnerable to forged-origin subprefix hijacks. See [GSG17] for details.

These measurements suggest that operators commonly misconfigure the maxLength attribute, and unwittingly open themselves up to forged-origin subprefix hijacks. That is, they are exposing a much larger attack surface for forged-origin hijacks than necessary.

5. Recommendations about Minimal ROAs and maxLength

Operators SHOULD use "minimal ROAs" whenever possible. A minimal ROA contains only those IP prefixes that are actually originated by an AS in BGP and no other IP prefixes. (See Section 3 for an example.)

In general, except in some special cases, operators SHOULD avoid using the maxLength attribute in their ROAs, since its inclusion will usually make the ROA non-minimal.

One such exception maybe when all more specific prefixes permitted by the maxLength are actually announced by the AS in the ROA. Another exception is where: (a) the maxLength is substantially larger compared to the specified prefix length in the ROA, and (b) a large number of more specific prefixes in that range are announced by the AS in the ROA. This case should occur rarely in practice (if at all). Operator discretion is necessary in this case.

This practice requires no changes to the RPKI specification and need not increase the number of signed ROAs in the RPKI, because ROAs already support lists of IP prefixes [RFC6482]. See also [GSG17] for further discussion of why this practice will have minimal impact on the performance of the RPKI ecosystem.

Operators that have existing ROAs published in the RPKI system SHOULD perform a review of such objects, especially where they make use of the maxLength attribute, to ensure that the set of included prefixes is "minimal" with respect to the current BGP origination and routing policies, and replace the published ROAs as necessary. Such an exercise SHOULD be repeated whenever the operator makes changes to either policy.

5.1. Facilitating Ad-hoc Routing Changes and DDoS Mitigation

Operational requirements may require that a route for an IP prefix be originated on an ad-hoc basis, with little or no prior warning. An example of such a situation arises when an operator wishes to make use of DDoS mitigation services that use BGP to redirect traffic via a "scrubbing center".

In order to ensure that such ad-hoc routing changes are effective, there should exist a ROA validating the new route. However a difficulty arises due to the fact that newly created objects in the RPKI are made visible to relying parties considerably more slowly than routing updates in BGP.

Ideally, it would not be necessary to pre-create the ROA which validates the ad-hoc route; instead create it "on-the-fly" as required. However, this is practical only if the latency imposed by the propagation of RPKI data is guaranteed to be within acceptable limits in the circumstances. For time-critical interventions such as responding to a DDoS attack, this is unlikely to be the case.

Thus, the ROA in question will usually need to be created well in advance of the routing intervention, but such a ROA will be non-minimal, since it includes an IP prefix that is sometimes (but not always) originated in BGP.

In this case, the ROA SHOULD include only:

- (1) the set of IP prefixes that are always originated in BGP, and
- (2) the set of IP prefixes that are sometimes, but not always, originated in BGP.

The ROA SHOULD NOT include any IP prefixes that the operator knows will not be originated in BGP. In general, the ROA SHOULD NOT make use of the maxLength attribute unless doing so has no impact on the set of included prefixes.

The running example is now extended to illustrate one situation where it is not possible to issue a minimal ROA.

Consider the following scenario prior to the deployment of RPKI. Suppose AS 64496 announced 192.168.0.0/16 and has a contract with a Distributed Denial of Service (DDoS) mitigation service provider that holds AS 64500. Further, assume that the DDoS mitigation service contract applies to all IP addresses covered by 192.168.0.0/22. When a DDoS attack is detected and reported by AS 64496, AS 64500 immediately originates 192.168.0.0/22, thus attracting all the DDoS traffic to itself. The traffic is scrubbed at AS 64500 and then sent back to AS 64496 over a backhaul data link. Notice that, during a DDoS attack, the DDoS mitigation service provider AS 64500 originates a /22 prefix that is longer than AS 64496's /16 prefix, and so all the traffic (destined to addresses in 192.168.0.0/22) that normally goes to AS 64496 goes to AS 64500 instead. In some deployments, the origination of the /22 route is performed by AS 64496 and announced only to AS 64500, which then announces transit for that prefix. This variation does not change the properties considered here.

First, suppose the RPKI only had the minimal ROA for AS 64496, as described in Section 3. But if there is no ROA authorizing AS 64500 to announce the /22 prefix, then the DDoS mitigation (and traffic scrubbing) scheme would not work. That is if AS 64500 originates the /22 prefix in BGP during DDoS attacks, the announcement would be invalid [RFC6811].

Therefore, the RPKI should have two ROAs: one for AS 64496 and one for AS 64500.

ROA:(192.168.0.0/16, 192.168.225.0/24, AS 64496)

ROA:(192.168.0.0/22, AS 64500)

Neither ROA uses the maxLength attribute. But the second ROA is not "minimal" because it contains a /22 prefix that is not originated by anyone in BGP during normal operations. The /22 prefix is only originated by AS 64500 as part of its DDoS mitigation service during a DDoS attack.

Notice, however, that this scheme does not come without risks. Namely, all IP addresses in 192.168.0.0/22 are vulnerable to a forged-origin subprefix hijack during normal operations, when the /22 prefix is not originated. (The hijacker AS 64511 would send the BGP announcement "192.168.0.0/22: AS 64511, AS 64500", falsely claiming that AS 64511 is a neighbor of AS 64500 and falsely claiming that AS 64500 originates 192.168.0.0/22.)

In some situations, the DDoS mitigation service at AS 64500 might want to limit the amount of DDoS traffic that it attracts and scrubs. Suppose that a DDoS attack only targets IP addresses in 192.168.0.0/24. Then, the DDoS mitigation service at AS 64500 only wants to attract the traffic designated for the /24 prefix that is under attack, but not the entire /22 prefix. To allow for this, the RPKI should have two ROAs: one for AS 64496 and one for AS 64500.

ROA:(192.168.0.0/16, 192.168.225.0/24, AS 64496)

ROA:(192.168.0.0/22-24, AS 64500)

The second ROA uses the maxLength attribute because it is designed to explicitly enable AS 64500 to originate *any* /24 subprefix of 192.168.0.0/22.

As before, the second ROA is not "minimal" because it contains prefixes that are not originated by anyone in BGP during normal operations. As before, all IP addresses in 192.168.0.0/22 are vulnerable to a forged-origin subprefix hijack during normal operations, when the /22 prefix is not originated.

The use of maxLength in this second ROA also comes with additional risk. While it permits the DDoS mitigation service at AS 64500 to originate prefix 192.168.0.0/24 during a DDoS attack in that space, it also makes the *other* /24 prefixes covered by the /22 prefix (i.e., 192.168.1.0/24, 192.168.2.0/24, 192.168.3.0/24) vulnerable to a forged-origin subprefix attacks.

5.2. Defensive de-aggregation in response to prefix hijacks

In responding to certain classes of prefix hijack, in particular, the forged-origin subprefix hijack described above, it may be desirable for the victim to perform "defensive de-aggregation". I.e., begin originating more-specific prefixes in order to compete with the hijacked route for selection as the best path in networks that are not performing RPKI-based route origin validation [RFC6811].

In some topologies, where at least one AS on every path between the victim and hijacker filters ROV invalid prefixes, it may be the case that the existence of a minimal ROA issued by the victim prevents the defensive more-specific prefixes being propagated to the networks topologically close to the attacker, thus hampering the effectiveness of this response.

Nevertheless, this document recommends that where possible, network operators publish minimal ROAs even in the face of this risk. This is because:

- * Minimal ROAs offer the best possible protection against the immediate impact of such an attack, rendering the need for such a response less likely;
- * Increasing ROV adoption by network operators will, over time, decrease the size of the neighborhoods in which this risk exists; and
- * Other methods for reducing the size of such neighborhoods are available to potential victims, such as establishing direct eBGP adjacencies with networks from whom the defensive routes would otherwise be hidden.

6. Considerations for RTBH Filtering Scenarios

Considerations related to ROAs and origin validation [RFC6811] for the case of destination-based Remote Triggered Black Hole (RTBH) filtering are addressed here. In RTBH filtering, highly specific prefixes (greater than /24 in IPv4 and greater than /48 in IPv6; possibly even /32 (IPv4) and /128 (IPv6)) are announced in BGP. These announcements are tagged with a BLACKHOLE Community [RFC7999]. It is obviously not desirable to use a large maxLength or include any such highly specific prefixes in the ROAs to accommodate destination-based RTBH filtering, for the reasons set out above.

As a result, RPKI-based route origin validation [RFC6811] is a poor fit for the validation of RTBH routes. Specification of new procedures to address this use case through the use of the RPKI is outside the scope of this document.

Therefore:

- * Operators SHOULD NOT create non-minimal ROAs (either by creating additional ROAs, or through the use of maxLength) for the purpose of advertising RTBH routes; and
- * Operators providing a means for operators of neighboring autonomous systems to advertise RTBH routes via BGP MUST NOT make the creation of non-minimal ROAs a pre-requisite for its use.

7. IANA Considerations

This document includes no request to IANA.

8. Security Considerations

This document makes recommendations regarding the use of RPKI-based origin validation as defined in [RFC6811], and as such introduces no additional security considerations beyond those set out therein.

The recommendations set out in this document, in particular, those in Section 5, involve trade-offs between operational agility and security. Operators are encouraged to carefully review the issues highlighted in light of their specific operational requirements. Failure to do so could, in the worst case, result in a self-inflicted denial of service.

9. Acknowledgments

The authors would like to thank the following people for their review and contributions to this document: Omar Sagga and Aris Lambrianidis. Thanks are also due to Matthias Waehlich, Ties de Kock, and Amreesh Phokeer for comments and suggestions.

10. References

10.1. Normative References

- [RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <<https://www.rfc-editor.org/info/rfc1918>>.
- [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>>.
- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <<https://www.rfc-editor.org/info/rfc4271>>.
- [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>>.
- [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>>.

10.2. Informative References

- [GCHSS] Gilad, Y., Cohen, A., Herzberg, A., Schapira, M., and H. Shulman, "Are We There Yet? On RPKI's Deployment and Security", in NDSS 2017, February 2017, <<https://eprint.iacr.org/2016/1010.pdf>>.
- [GSG17] Gilad, Y., Sagga, O., and S. Goldberg, "Maxlength Considered Harmful to the RPKI", in ACM CoNEXT 2017, December 2017, <<https://eprint.iacr.org/2016/1015.pdf>>.

- [HARMFUL] Gilad, Y., Sagga, O., and S. Goldberg, "MaxLength Considered Harmful to the RPKI", 2017, <<https://eprint.iacr.org/2016/1015.pdf>>.
- [LSG16] Lychev, R., Shapira, M., and S. Goldberg, "Rethinking Security for Internet Routing", in Communications of the ACM, October 2016, <<http://cacm.acm.org/magazines/2016/10/207763-rethinking-security-for-internet-routing/>>.
- [RFC5737] Arkko, J., Cotton, M., and L. Vegoda, "IPv4 Address Blocks Reserved for Documentation", RFC 5737, DOI 10.17487/RFC5737, January 2010, <<https://www.rfc-editor.org/info/rfc5737>>.
- [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>>.
- [RFC6907] Manderson, T., Sriram, K., and R. White, "Use Cases and Interpretations of Resource Public Key Infrastructure (RPKI) Objects for Issuers and Relying Parties", RFC 6907, DOI 10.17487/RFC6907, March 2013, <<https://www.rfc-editor.org/info/rfc6907>>.
- [RFC7115] Bush, R., "Origin Validation Operation Based on the Resource Public Key Infrastructure (RPKI)", BCP 185, RFC 7115, DOI 10.17487/RFC7115, January 2014, <<https://www.rfc-editor.org/info/rfc7115>>.
- [RFC7999] King, T., Dietzel, C., Snijders, J., Doering, G., and G. Hankins, "BLACKHOLE Community", RFC 7999, DOI 10.17487/RFC7999, October 2016, <<https://www.rfc-editor.org/info/rfc7999>>.
- [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>>.

Authors' Addresses

Yossi Gilad
Hebrew University of Jerusalem
Rothburg Family Buildings, Edmond J. Safra Campus
Jerusalem 9190416
Israel
Email: yossigi@cs.huji.ac.il

Sharon Goldberg
Boston University
111 Cummington St, MCS135
Boston, MA 02215
United States of America
Email: goldbe@cs.bu.edu

Kotikalapudi Sriram
USA National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD 20899
United States of America
Email: kotikalapudi.sriram@nist.gov

Job Snijders
Fastly
Amsterdam
Netherlands
Email: job@fastly.com

Ben Maddison
Workonline Communications
114 West St
Johannesburg
2196
South Africa
Email: benm@workonline.africa

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 8 September 2022

C. Martinez
LACNIC
G. Michaelson
T. Harrison
APNIC
T. Bruijnzeels
NLnet Labs
R. Austein
Dragon Research Labs
7 March 2022

RPKI Signed Object for Trust Anchor Key
draft-ietf-sidrops-signed-tal-09

Abstract

A Trust Anchor Locator (TAL) is used by Relying Parties (RPs) in the Resource Public Key Infrastructure (RPKI) to locate and validate a Trust Anchor (TA) Certification Authority (CA) certificate used in RPKI validation. This document defines an RPKI signed object for a Trust Anchor Key (TAK), that can be used by a TA to signal the location(s) of the accompanying CA certificate for the current key to RPs, as well as the successor key and the location(s) of its CA certificate. This object helps to support planned key rolls without impacting RPKI validation.

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 8 September 2022.

Copyright Notice

Copyright (c) 2022 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. Requirements Notation	3
2. Overview	3
3. TAK Object Definition	4
3.1. The TAK Object Content Type	4
3.2. The TAK Object eContent	4
3.2.1. TAKey	4
3.2.2. TAK	5
3.3. TAK Object Validation	5
4. TAK Object Generation and Publication	6
5. Relying Party Use	7
6. Maintaining Multiple TA Keys	9
7. Performing TA Key Rolls	10
7.1. Phase 1: Add a TAK for Key 'A'	10
7.2. Phase 2: Add a Key 'B'	10
7.3. Phase 3: Update TAL to point to 'B'	11
7.4. Phase 4: Remove Key 'A'	11
8. Deployment Considerations	11
9. Security Considerations	12
10. IANA Considerations	13
10.1. OID	13
10.2. File Extension	13
10.3. Module Identifier	13
11. Implementation Status	13
11.1. APNIC	14
12. Revision History	14
13. Acknowledgments	15
14. References	15
14.1. Normative References	15
14.2. Informative References	16
Appendix A. ASN.1 Module	16
Authors' Addresses	17

1. Requirements Notation

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. Overview

A TAL [RFC8630] is used by an RP in the RPKI to locate and validate TA CA certificates used in RPKI validation. However, until now there has been no in-band way of notifying RPs of updates to a TAL. In-band notification means that TAs can be more confident of RPs being aware of key roll operations.

This document defines a new RPKI signed object that can be used to document the location(s) of the TA CA certificate for the current TA key, as well as the value of the successor key and the location(s) of its TA CA certificate. This allows RPs to be notified automatically of such changes, and enables TAs to stage a successor key so that planned key rolls can be performed without risking the invalidation of the RPKI tree under the TA. We call this object the Trust Anchor Key (TAK) object.

When RPs are first bootstrapped, they use a TAL to discover the key and location(s) of the CA certificate for a TA. The RP can then retrieve and validate the CA certificate, and subsequently validate the manifest [RFC6486] and CRL published by that TA (section 5 of [RFC6487]). However, before processing any other objects it will first validate the TAK object, if present. If the TAK object lists only the current key, then the RP continues processing as per normal. If the TAK object includes a successor key, the RP starts an acceptance timer, and then continues processing as per normal. If, during the following validation runs up until the expiry of the acceptance timer, the RP has not observed any changes to the keys and certificate URLs listed in the TAK object, then the RP will fetch the successor key, update its local state with that key and its associated certification location(s), and continue processing using that key.

The primary motivation for this work is being able to migrate from a Hardware Security Module (HSM) produced by one vendor to one produced by another, where the first vendor does not support exporting keys for use by the second. There may be other scenarios in which key rollover is useful, though.

3. TAK Object Definition

The TAK object makes use of the template for RPKI digitally signed objects [RFC6488], which defines a Cryptographic Message Syntax (CMS) [RFC5652] wrapper for the content as well as a generic validation procedure for RPKI signed objects. Therefore, to complete the specification of the TAK object (see Section 4 of [RFC6488]), this document defines:

- * The OID (in Section 3.1) that identifies the signed object as being a TAK. (This OID appears within the eContentType in the encapContentInfo object, as well as the content-type signed attribute in the signerInfo object.)
- * The ASN.1 syntax for the TAK eContent, in Section 3.2.
- * The additional steps required to validate a TAK, in Section 3.3.

3.1. The TAK Object Content Type

This document requests an OID for the TAK object as follows:

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

The content of a TAK object is ASN.1 encoded using the Distinguished Encoding Rules (DER) [X.690], and is defined per the module in Appendix A.

3.2.1. TAKey

This structure defines a TA key, similarly to [RFC8630]. It contains a sequence of one or more URIs and a SubjectPublicKeyInfo.

3.2.1.1. certificateURIs

This field is equivalent to the URI section defined in section 2.2 of [RFC8630]. It MUST contain at least one CertificateURI element. Each CertificateURI element contains the IA5String representation of either an rsync URI [RFC5781], or an HTTPS URI [RFC7230].

3.2.1.2. subjectPublicKeyInfo

This field contains a SubjectPublicKeyInfo (section 4.1.2.7 of [RFC5280]) in DER format [X.690].

3.2.2. TAK

3.2.2.1. version

The version number of the TAK object MUST be 0.

3.2.2.2. current

This field contains the TA key of the repository in which the TAK object is published.

3.2.2.3. predecessor

This field contains the TA key that was in use for this TA immediately prior to the current TA key, if applicable.

3.2.2.4. successor

This field contains the TA key to be used in place of the current key, after expiry of the relevant acceptance timer.

3.3. TAK Object Validation

To determine whether a TAK object is valid, the RP MUST perform the following checks in addition to those specified in [RFC6488]:

- * The eContentType OID matches the OID described in Section 3.1.
- * The TAK object appears as the product of a TA CA certificate (i.e. the TA CA certificate is itself the issuer of the EE certificate of the TAK object).
- * The TA CA has published only one TAK object in its repository for this key, and this object appears on the manifest as the only entry using the ".tak" extension (see [RFC6481]).
- * The EE certificate of this TAK object describes its Internet Number Resources (INRs) using the "inherit" attribute.
- * The decoded TAK content conforms to the format defined in Section 3.2.

- * The SubjectPublicKeyInfo value of the current TA key in the TAK object matches that of the TA CA certificate used to issue the EE certificate of the TAK object.

If any of these checks does not succeed, the RP MUST ignore the TAK object, and proceed as though it were not listed on the manifest.

The RP is not required to compare its current set of certificateURIs for the current key with those listed in the TAK object. The RP MAY alert the user that these sets of certificateURIs do not match, with a view to the user manually updating the set of certificateURIs in their configuration. The RP MUST NOT automatically update its configuration to use these certificateURIs in the event of inconsistency, though, because migration of users to new certificateURIs should happen by way of the successor key process.

4. TAK Object Generation and Publication

A TA MAY choose to use TAK objects to communicate its current, predecessor, and successor keys. If a TA chooses to use TAK objects, then it SHOULD generate and publish TAK objects under each of its keys.

A non-normative guideline for naming this object is that the filename chosen for the TAK object 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 ".tak" MUST be used to denote the object as a TAK.

In order to generate a TAK object, the TA MUST perform the following actions:

- * The TA MUST generate a key pair for a "one-time-use" EE certificate to use for the TAK.
- * The TA MUST generate a one-time-use EE certificate for the TAK.
- * 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 TAK as an object URL.

- * 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.
- * This EE certificate MUST have a "notBefore" time that matches or predates the moment that the TAK will be published.
- * This EE certificate MUST have a "notAfter" time that reflects the intended duration for which this TAK will be published. If the EE certificate for a TAK object is expired, it MUST no longer be published, but it MAY be replaced by a newly generated TAK object with equivalent content and an updated "notAfter" time.
- * The current TA key for the TAK MUST match that of the TA CA certificate under which the TAK was issued.

5. Relying Party Use

Relying Parties MUST keep a record of the current key for each configured TA, as well as the URI(s) where the CA certificate for this key may be retrieved. This record is typically bootstrapped by the use of a pre-configured (and unsigned) TAL file [RFC8630].

When performing top-down validation, RPs MUST first validate and process the TAK object for its current known key, by performing the following steps:

- * A CA certificate is retrieved and validated from the known URIs as described in sections 3 and 4 of [RFC8630].
- * The manifest and CRL for this certificate are then validated as described in [RFC6487] and [RFC6486].
- * The TAK object, if present, is validated as described in Section 3.3.

If the TAK object includes a successor key, then the RP must verify the successor key by doing the following:

- * performing top-down validation using the successor key, in order to validate the TAK object for the successor TA;
- * ensuring that a valid TAK object exists for the successor TA;

- * ensuring that the successor TAK object's current key matches the initial TAK object's successor key; and
- * ensuring that the successor TAK object's predecessor key matches the initial TAK object's current key.

If any of these steps fails, then the successor key has failed verification.

If the successor key passes verification, and the RP has not seen that successor key on the previous successful validation run for this TA, then the RP:

- * sets an acceptance timer of 30 days for this successor key for this TA;
- * cancels the existing acceptance timer for this TA (if applicable); and
- * continues standard top-down validation as described in [RFC6487] using the current key.

If the successor key passes verification, and the RP has seen that successor key on the previous successful validation run for this TA:

- * if the relevant acceptance timer has not expired, the RP continues standard top-down validation using the current key;
- * otherwise, the RP updates its current known key details for this TA to be those of the successor key, and then begins top-down validation again using the successor key.

If the successor key does not pass verification, or if the TAK object does not include a successor key, the RP cancels the existing acceptance timer for this TA (if applicable).

An RP MUST NOT use a successor key for top-down validation outside of the process described above, except for the purpose of testing that the new key is working correctly. This allows a TA to publish a successor key for a period of time, allowing RPs to test it, while still being able to rely on RPs using the current key for their production RPKI operations.

A successor key may have the same SubjectPublicKeyInfo value as the current key: this will be the case where a TA is updating the certificateURIs for that key.

6. Maintaining Multiple TA Keys

Although an RP that can process TAK objects will only ever use one key for validation (either the current key, or the successor key, once the relevant acceptance timer has expired), an RP that cannot process TAK objects will continue to use the key details per its TAL (or equivalent manual configuration) indefinitely. As a result, even when a TA is using a TAK object in order to migrate clients to a new key, the TA may have to maintain the previous key for a period of time alongside the new key in order to ensure continuity of service for older clients.

For each TA key that a TA is maintaining, the signed material for these keys MUST be published under different directories in the context of the 'id-ad-caRepository' and 'id-ad-rpkiManifest' Subject Information Access descriptions contained on the CA certificates [RFC6487]. Publishing objects under the same directory is potentially confusing for RPs, and could lead to object invalidity in the event of file name collisions.

Also, the CA certificates for each maintained key, and the contents published by each key, MUST be equivalent (except for the TAK object). In other words, for the purposes of RPKI validation, it MUST NOT make a difference which of the keys is used as a starting point.

This means that the IP and AS resources contained on all current CA certificates for the maintained TA keys MUST be the same. Furthermore, for any delegation of IP and AS resources to a child, the TA MUST have an equivalent CA certificate published under each of its keys. Any updates in delegations MUST be reflected under each of its keys. A TA SHOULD NOT publish any other objects besides a CRL, a Manifest, a single TAK object, and any number of CA certificates for delegation to child CAs.

If a TA uses a single remote publication server for its keys, per [RFC8181], then it MUST include all <publish/> and <withdraw/> PDUs for the products of each of its keys in a single query, in order to ensure that they will reflect the same content at all times.

If a TA uses multiple publication servers, then it is by definition inevitable that the content of different keys will be out of sync at times. In such cases, the TA SHOULD ensure that the duration of these moments are limited to the shortest possible time. Furthermore, the following should be observed:

- * In cases where a CA certificate is revoked completely, or replaced by a certificate with a reduced set of resources, these changes will not take effect fully until all the relevant repository publication points have been updated. Given that TA key operations are normally performed infrequently, this is unlikely to be a problem: if the revocation or shrinking of an issued CA certificate is staged for days/weeks, then experiencing a delay of several minutes for the repository publication points to be updated is fairly insignificant.
- * In cases where a CA certificate is replaced by a certificate with an extended set of resources, the TA MUST inform the receiving CA only after all of its repository publication points have been updated. This ensures that the receiving CA will not issue any products that could be invalid if an RP uses a TA key just before the CA certificate was due to be updated.

Finally, note that the publication locations of CA certificates for delegations to child CAs under each key will be different, and therefore the Authority Information Access 'id-ad-caIssuers' values (section 4.8.7 of [RFC6487]) on certificates issued by the child CAs may not be as expected when performing top-down validation, depending on the TA key that is used. However, these values are not critical to top-down validation, so RPs performing such validation MUST NOT reject a certificate simply because this value is not as expected.

7. Performing TA Key Rolls

In this section we will describe how present-day RPKI TAs that use only one key pair, and that do not use TAK objects, can use a TAK object to perform a planned key roll.

7.1. Phase 1: Add a TAK for Key 'A'

Before adding a successor key, a TA may want to confirm that it can maintain a TAK object for its current key only. We will refer to this key as key 'A' throughout this section.

7.2. Phase 2: Add a Key 'B'

The TA can now generate a new key pair for key 'B'. This key MUST now be used to create a new CA certificate for this key, and to issue equivalent CA certificates for delegations to child CAs, as described in Section 6.

At this point, the TA can also construct a new TAL file [RFC8630] for key 'B', and test locally that the validation outcome for the new key is equivalent to that of the other current key(s).

When the TA is certain that both keys are equivalent, and wants to initiate the migration from 'A' to 'B', it issues a new TAK object under key 'A', with key 'A' as the current key for that object, key 'B' as the successor key, and no predecessor key. It also issues a TAK object under key 'B', with key 'B' as the current key for that object, key 'A' as the predecessor key, and no successor key.

Once this has happened, RP clients will start seeing the new key and setting acceptance timers accordingly.

7.3. Phase 3: Update TAL to point to 'B'

At about the time that the TA expects clients to start setting key 'B' as the current key, the TA must release a new TAL file for key 'B'. It SHOULD use a different set of URIs in the TAL compared to the TAK file, so that the TA can learn the proportion of RPs that can successfully validate and use the updated TAK objects.

To support RPs that do not take account of TAK objects, the TA should continue operating key 'A' for a period of time after the expected migration of clients to 'B'. The length of that period of time is a local policy matter for that TA: it might operate the key until no clients are attempting to validate using it, for example.

7.4. Phase 4: Remove Key 'A'

The TA SHOULD now remove all content from the repository used by key 'A', and destroy the private key for key 'A'. RPs attempting to rely on a TAL for key 'A' from this point will not be able to perform RPKI validation for the TA, and will have to update their local state manually, by way of a new TAL file.

8. Deployment Considerations

Including TAK objects while RPs do not support this standard will result in those RPs rejecting these objects. It is not expected that this will result in the invalidation of any other object under a Trust Anchor.

The mechanism introduced here can only be relied on once a majority of RPs support it. Defining when that moment arrives is something that cannot be established at the time of writing this document. The use of unique URIs for keys in TAK objects, different from those used for the corresponding TAL files, should help TAs understand the proportion of RPs that support this mechanism.

Some RPs may purposefully not support this mechanism: for example, they may be implemented or configured such that they are unable to update local current key data. TAs should take this into consideration when planning key rollover. However, these RPs would ideally still notify their operators of planned key rollovers, so that the operator could update the relevant configuration manually.

9. Security Considerations

A TA needs to consider the length of time for which it will maintain previously-current keys and their associated repositories. An RP that is seeded with old TAL data will run for 30 days using the previous key before migrating to the next key, due to the acceptance timer requirements, and this 30-day delay applies to each new key that has been issued since the old TAL data was initially published. It may be better in these instances to have the old publication URLs simply fail to resolve, so that the RP reports an error to its operator and the operator seeds it with up-to-date TAL data immediately.

Once a TA has decided not to maintain a previously-current key and its associated repository, it needs to consider how to protect against an adversary gaining access to that key and its associated publication points in order to send invalid/incorrect data to RPs seeded with the TAL data for that key. One possible mitigation here is to reuse the TA CA certificate URLs from that TAL data for newer keys.

The use of acceptance timers means that an adversary that gains access to a TA's current key is not able to migrate RPs to a new key without delay. Although access to that key does permit arbitrary action within the corresponding TA (assuming that the adversary has control over the relevant publication points), being unable to migrate RPs to a new key means that it is possible for the TA operator to regain control over the key and the TA itself, such that it may not be necessary for all RPs to carry out manual reconfiguration.

If an adversary gains access to the key listed as the successor to a TA's current key (i.e. listed as the successor, but the acceptance timer period has not yet elapsed since it was listed), the TA operator can recover from this by simply removing the successor key from the TAK object.

In general, the risk of key compromise can be mitigated by the use of Hardware Security Modules (HSMs) by TAs, which will guard against theft of a private key, as well as operational processes to guard against (accidental) misuse of the keys in an HSM by operators.

Alternate models of TAL update exist and can be complementary to this mechanism. For example, TAs can liaise directly with validation software developers to include updated and reissued TAL files in new code releases, and use existing code update mechanisms in the RP community to distribute the changes.

10. IANA Considerations

10.1. OID

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

Decimal	Description	References
TBD	Trust Anchor Key	[section 3.1]

10.2. File Extension

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

Extension	RPKI Object	References
.tak	Trust Anchor Key	[this document]

10.3. Module Identifier

IANA is asked to register an object identifier for one module identifier in the "SMI Security for S/MIME Module Identifier (1.2.840.113549.1.9.16.0)" registry as follows:

Decimal	Description	References
TBD	Trust Anchor Key	[section 3.1]

* Description: RPKISignedTrustAnchorList-2021

* OID: 1.2.840.113549.1.9.16.0.TBD

* Specification: [this document]

11. Implementation Status

NOTE: Please remove this section and the reference to RFC 7942 prior to publication 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 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".

11.1. APNIC

- * Responsible Organization: Asia-Pacific Network Information Centre
- * Location: <https://github.com/APNIC-net/rpki-signed-tal-demo>
- * Description: A proof-of-concept for relying party TAK usage.
- * Level of Maturity: This is a proof-of-concept implementation.
- * Coverage: This implementation includes all of the features described in version 08 of this specification. The repository includes a link to various test TALs that can be used for testing TAK scenarios, too.
- * Contact Information: Tom Harrison, tomh@apnic.net

12. Revision History

- 03 - Last draft under Tim's authorship.
- 04 - First draft with George's authorship. No substantive revisions.
- 05 - First draft with Tom's authorship. No substantive revisions.
- 06 - Rob Kisteleki's critique.
- 07 - Switch to two-key model.

08 - Keepalive.

09 - Acceptance timers, predecessor keys, no long-lived CRL/MFT.

13. Acknowledgments

The authors wish to thank Martin Hoffmann for a thorough review of this document, and Russ Housley for reviewing the ASN.1 definitions and providing a new module for the TAK object.

14. References

14.1. 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>>.
- [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>>.
- [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>>.
- [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>>.
- [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>>.
- [RFC8181] Weiler, S., Sonalker, A., and R. Austein, "A Publication Protocol for the Resource Public Key Infrastructure (RPKI)", RFC 8181, DOI 10.17487/RFC8181, July 2017, <<https://www.rfc-editor.org/info/rfc8181>>.
- [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>>.
- [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.

14.2. Informative References

- [RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <<https://www.rfc-editor.org/info/rfc5652>>.
- [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>>.

Appendix A. ASN.1 Module

This appendix includes the ASN.1 module for the TAK object.


```

<CODE BEGINS>
RPKISignedTrustAnchorList-2021
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
    pkcs9(9) smime(16) mod(0) TBD }

DEFINITIONS EXPLICIT TAGS ::=
BEGIN

IMPORTS

CONTENT-TYPE
  FROM CryptographicMessageSyntax-2009 -- in [RFC5911]
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
    pkcs9(9) smime(16) modules(0) id-mod-cms-2004-02(41) }

SubjectPublicKeyInfo
  FROM PKIX1Explicit-2009 -- in [RFC5912]
  { iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) id-mod(0)
    id-mod-pkix1-explicit-02(51) } ;

ct-signedTAL CONTENT-TYPE ::=
  { TYPE TAK IDENTIFIED BY
    id-ct-signedTAL }

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

CertificateURI ::= IA5String

TAKey ::= SEQUENCE {
  certificateURIs SEQUENCE SIZE (1..MAX) OF CertificateURI,
  subjectPublicKeyInfo SubjectPublicKeyInfo
}

TAK ::= SEQUENCE {
  version INTEGER DEFAULT 0,
  current TAKey,
  predecessor TAKey OPTIONAL,
  successor TAKey OPTIONAL
}

END
<CODE ENDS>

```

Authors' Addresses

Carlos Martinez
LACNIC
Email: carlos@lacnic.net
URI: <https://www.lacnic.net/>

George G. Michaelson
Asia Pacific Network Information Centre
6 Cordelia St
South Brisbane
QLD 4101
Australia
Email: ggm@apnic.net

Tom Harrison
Asia Pacific Network Information Centre
6 Cordelia St
South Brisbane
QLD 4101
Australia
Email: tomh@apnic.net

Tim Bruijnzeels
NLnet Labs
Email: tim@nlnetlabs.nl
URI: <https://www.nlnetlabs.nl/>

Rob Austein
Dragon Research Labs
Email: sra@hactrn.net

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 3, 2020

R. Bush
Internet Initiative Japan & Arrcus
K. Patel
Arrcus
July 2, 2019

Origin Validation Signaling
draft-ymbk-sidrops-ov-signal-03

Abstract

Within a trust boundary, e.g. an operator's PoP, it may be useful to have only a few central devices do full Origin Validation using the Resource Public Key Infrastructure, and be able to signal to an internal sender that a received route fails Origin Validation. E.g. route reflectors could perform Origin Validation for a cluster and signal back to a sending client that it sent an invalid route. Routers capable of sending and receiving this signal can use the extended community described in [RFC8097].

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 <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 January 3, 2020.

Copyright Notice

Copyright (c) 2019 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.

1. Introduction

Within a routing trust boundary, e.g. an operator's Point of Presence (PoP), it may not be desirable or necessary for all routers to perform Origin Validation using the Resource Public Key Infrastructure (RPKI) per [RFC6811]. A good example is route reflectors (see [RFC4456]).

An RPKI-enabled device, an Evaluator, SHOULD signal receipt of an Invalid route back to the sender by announcing that route back to the sender marked with the BGP Prefix Origin Validation State Extended Community as defined in [RFC8097] with a last octet having the value 2, meaning "Invalid." In the rest of this document we take the liberty of calling it the "community."

We use the term "Sender" to refer to the router announcing routes to the device evaluating the Origin Validation of the announcements. Beware that the Sender receives signaling back from the Evaluator, which can be somewhat confusing.

We use the term "Evaluator" to describe the device receiving routing announcements from senders, applying RPKI-based Origin Validation, and possibly signaling route Invalidity back to the sender(s).

2. Suggested Reading

It is assumed that the reader understands BGP, [RFC4271], the RPKI, [RFC6480], RPKI-based Prefix Validation, [RFC6811], and the BGP Prefix Origin Validation State Extended Community as described in [RFC8097].

3. Trust Boundary

As a general rule, we discourage 'outsourcing trust,' i.e. letting others make security decisions for us. But there are operational environments with a somewhat wide trust boundary, a single operator's PoP for example.

This is not outsourcing trust; this is remote decision making. It is not letting a third party make the decision; it is simply doing it on a different computer. It's trust in a distributed system, where what is (sometimes) called the Policy Decision Point is not the same as the Policy Enforcement Point.

As described in [RFC7115], a PoP might have a single RPKI Cache, hence all trust is vested in it. So it is reasonable that routers in that PoP could share Origin Validation results instead of each doing full validation.

An [RFC4456] Route Reflector Cluster is an obvious candidate for this approach. The route reflector(s) would perform Origin Validation and signal an Invalid route back to the sending client.

[RFC8097] provides the obvious signaling mechanism, the BGP Prefix Origin Validation State Extended Community. The device performing OV SHOULD signal back to the sender by announcing the offending prefix marked with the extended community with the last octet having the value 2, indicating an Invalid route.

4. The OV Signaling Capability

Unfortunately, the router sending the Invalid announcement is not normally expecting to receive it back. Therefore, both parties MUST agree on this feature by using a BGP Capability [RFC5492].

To advertise the OV Signaling Capability to a peer, a BGP speaker uses BGP Capabilities Advertisement [RFC5492]. By advertising the OV Signaling Capability to a peer, a BGP speaker conveys that it is able to send, receive, and properly handle OV Signaling using the community.

A peer which does not advertise this capability MUST NOT send OV Signaling, and BGP OV Signaling MUST NOT be sent to it.

The OV Signaling Capability is a new BGP Capability defined with Capability code [TBD] and Capability length 0.

5. Recommended Action

This section assumes that the OV Signaling Capability has been negotiated by the sending and receiving routers.

An Evaluating device which performs Origin Validation on a route received from a capable sender and finds a prefix with a particular origin AS to be Invalid (in the [RFC6811] sense), MUST announce that prefix back to the sending router from which it was received with the Invalid origin AS and the addition of the community with the last octet being 2.

A sender receiving the returned prefix announcement so marked MUST treat it the way it would treat an Invalid origin that it itself detected. It should withdraw all routes it had announced to that prefix with the Invalid origin AS. This includes withdrawing any instances of additional paths with that origin AS advertised under [RFC7911].

For a sender to properly evaluate the community returned by the evaluator, the sender MUST recognize the community before loop detection. This is a change to the Phase 2 Route Selection process of [RFC4271] Section 9.1.2.

If a sender originally received the Invalid route from an evaluator within its trust boundary with which it has negotiated the OV Signaling Capability, it MAY also propagate that signal to the original sender.

6. Security Considerations

As with all communities which cause semantic change, this use of the community may be abused as an attack vector. Therefore the operator MUST configure their incoming external border to strip the community.

As the BGP sessions are already established using whatever channel security the operator chooses or not, this change specifies no additional channel or object security. Of course, the BGP transport should be protected for integrity and authentication. TCP-MD5 [RFC2385] is available on almost all platforms. If more modern methods are available, they should be used.

Outsourcing security is usually considered bad policy. Section Section 3 above discusses why that is not really the case here.

Otherwise, this document does not create security considerations beyond those of [RFC6811].

7. IANA Considerations

This document requests the IANA assign the "OV Signaling Capability" to the BGP Capabilities described in Section 2.1 in the "Capability Codes" registry's "IETF Review" range [RFC8126].. This document is the reference for the new capability.

8. Acknowledgments

Thanks to Steve Bellovin for a serious security review, and Rob Austein for a useful security snark.

9. References

9.1. 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>>.
- [RFC2385] Heffernan, A., "Protection of BGP Sessions via the TCP MD5 Signature Option", RFC 2385, DOI 10.17487/RFC2385, August 1998, <<http://www.rfc-editor.org/info/rfc2385>>.
- [RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)", RFC 4456, DOI 10.17487/RFC4456, April 2006, <<http://www.rfc-editor.org/info/rfc4456>>.
- [RFC5492] Scudder, J. and R. Chandra, "Capabilities Advertisement with BGP-4", RFC 5492, DOI 10.17487/RFC5492, February 2009, <<http://www.rfc-editor.org/info/rfc5492>>.
- [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>>.
- [RFC7115] Bush, R., "Origin Validation Operation Based on the Resource Public Key Infrastructure (RPKI)", BCP 185, RFC 7115, DOI 10.17487/RFC7115, January 2014, <<http://www.rfc-editor.org/info/rfc7115>>.
- [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>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<http://www.rfc-editor.org/info/rfc8126>>.

9.2. Informative References

- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <<http://www.rfc-editor.org/info/rfc4271>>.
- [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>>.

Authors' Addresses

Randy Bush
Internet Initiative Japan & Arrcus
5147 Crystal Springs
Bainbridge Island, Washington 98110
US

Email: randy@psg.com

Keyur Patel
Arrcus
2077 Gateway Place, Suite #250
San Jose, CA 95119
United States of America

Email: keyur@arrcus.com