Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 24 September 2025

A. Azimov
Yandex
E. Bogomazov
Qrator Labs
R. Bush
IIJ & Arrcus
K. Patel
Arrcus
J. Snijders

K. Sriram
USA NIST

23 March 2025

BGP AS_PATH Verification Based on Autonomous System Provider Authorization (ASPA) Objects
draft-ietf-sidrops-aspa-verification-22

Abstract

This document describes procedures that make use of Autonomous System Provider Authorization (ASPA) objects in the Resource Public Key Infrastructure (RPKI) to verify the Border Gateway Protocol (BGP) AS_PATH attribute of advertised routes. This AS_PATH verification enhances routing security by adding means to detect and mitigate route leaks and AS_PATH manipulations.

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 24 September 2025.

Table of Contents

1. Introduction 3
2. Requirements Language 4
3. Terminology and List of Acronyms 4
4. ASPA Registration Recommendations 4
5. Provider Authorization Function 5
6. AS_PATH Verification 6
6.1. Principles 7
6.2. Algorithm for Upstream Paths 9
6.3. Algorithm for Downstream Paths 9
6.4. Mitigation Policy 10
7. Deployment Recommendations 10
7.1. ASPA Verification Examples 10
7.2. Application of Verification Procedures 10
7.3. BGP Roles 11
7.4. Complex Peering Relationships 11
7.5. Logging 11
8. Security Considerations 11
8.1. Incongruence in IPv4 and IPv6 Connectivity 11
8.2. Correctness of the ASPA 12
8.3. Manipulation of AS_PATH by Provider 12
8.4. Manipulating AS_PATH Prepends 12
9. Relation to Other Technologies 12
9.1. ROA 12
9.2. BGPsec 13
9.3. Peerlock 13
9.4. Only to Customer (OTC) Attribute 13
10. IANA Considerations 13
11. Implementation Status 14
12. References 14
12.1. Normative References 15
12.2. Informative References 16
Appendix A. Acknowledgments 18

Appendix B. Properties and Early Adoption Benefits 18
Authors' Addresses 19

1. Introduction

The Border Gateway Protocol (BGP) as originally designed is known to be vulnerable to prefix (or route) hijacks and BGP route leaks [RFC7908]. Some existing BGP extensions can partially solve these problems. For example, Resource Public Key Infrastructure (RPKI) based route origin validation (RPKI-ROV) [RFC6480] [RFC6482] [RFC6811] [RFC9319] can be used to detect and filter accidental mis-originations. [RFC9234] or [I-D.ietf-grow-route-leak-detection-mitigation] can be used to detect and mitigate accidental route leaks. While RPKI-ROV can prevent accidental prefix hijacks, malicious forged-origin prefix hijacks can still occur [RFC9319]. RFC9319 includes some recommendations for reducing the attack surface for forged-origin prefix hijacks.

This document describes procedures that make use of Autonomous System Provider Authorization (ASPA) objects [I-D.ietf-sidrops-aspa-profile] in the RPKI to verify properties of the BGP AS_PATH attribute of advertised routes. ASPA-based AS_PATH verification provides detection and mitigation of route leaks. It also provides protection, to some degree, against prefix hijacks with forged-origin or forged-path-segment (Appendix B). These new ASPA-based procedures automatically detect such anomalous AS_PATHs in BGP Updates that are advertised between ASes.

Both route leaks and hijacks have similar effects on ISP operations. They redirect traffic and can result in denial of service (DoS), eavesdropping, increased latency, and packet loss. The level of risk, however, depends significantly on the extent of propagation of the anomalies. For example, a route leak or hijack that is propagated only to customers may cause bottlenecking within an ISP's customer cone, but if the anomaly propagates through lateral (i.e., non-transit) peers or transit providers, then the ill effects will likely be amplified and may be experienced worldwide.

The ability to constrain the propagation of BGP anomalies to transit providers and lateral peers -- without requiring support from the source of the anomaly (which is critical if the source has malicious intent) -- should significantly improve the robustness of the global inter-domain routing system.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terminology and List of Acronyms

The following terms are used with special meanings.

Route is ineligible: The term has the same meaning as in [RFC4271], i.e., "route is ineligible to be installed in Loc-RIB and will be excluded from the next phase of route selection."

CAS: Customer AS ([I-D.ietf-sidrops-aspa-profile], Section 1).

PAS: Provider AS ([I-D.ietf-sidrops-aspa-profile], Section 1).

SPAS: Set of Provider ASes ([I-D.ietf-sidrops-aspa-profile], Section 3).

For path verification purposes in this document, the peering relationships an AS can have in relation to a neighbor AS are Customer, Provider, Peer, Route Server (RS), RS-client, and Complex. These peering relationships are defined in [RFC9234]. All peering relationships are defined locally.

4. ASPA Registration Recommendations

An AS SHOULD register an ASPA. An AS MUST list in its SPAS the union of all its Provider AS(es) and non-transparent RS AS(es) at which it is an RS-client. An AS MUST include a Provider AS in its SPAS regardless of whether it provides connectivity for only IPv4 or only IPv6 or both.

In the Complex relationship case (Section 3 and [RFC9234]), an AS MUST include the neighbor AS in its SPAS if the neighbor plays Provider role for all or a subset of received or sent prefixes. Thus, if two ASes are exporting both customer and non-customer routes to each other, each AS registers its ASPA including the other AS in its SPAS.

The ASes on the boundary of an AS Confederation MUST register ASPAs using the Confederation’s global AS number (ASN) as the CAS.

An ASPA object showing only AS 0 as a provider AS is referred to as an AS0 ASPA. A non-transparent Route Server AS (RS AS) is one that includes its AS number in the AS_PATH. Registering as AS0 ASPA is a statement by the registering AS that it has no transit providers, and it is also not an RS-client at a non-transparent RS AS. If that statement is true, then the AS MUST register an AS0 ASPA.

Normally, a SPAS (see Section 3) is not expected to contain both an AS 0 and other Provider ASes, but an unexpected presence of AS 0 has no influence on the AS_PATH verification procedures (see Section 5, Section 6).

An AS SHOULD register a single ASPA object. A single ASPA record for an AS ought to prevent race conditions during ASPA updates that might affect prefix propagation. The CA software that provides hosting for ASPA records SHOULD support enforcement of this practice.

An AS may have providers that may be used on certain occasions, for an example in case of a DDoS attack. It is RECOMMENDED to add such providers in ASPA in advance, so there will be no race conditions between ASPA distribution and route propagation.

During a transition process between different certificate authority (CA) registries, the ASPA records SHOULD be kept identical in all relevant registries.

5. Provider Authorization Function

A CAS is expected to register a single ASPA listing all its Provider ASes (see Section 4). If a CAS has a single cryptographically valid ASPA, then the Union SPAS (U-SPAS) for the CAS equals to SPAS. In case a CAS has multiple cryptographically valid ASPAs, then the U-SPAS for the CAS is the union of AS listed in all SPAS of these ASPAs.

Let AS x and AS y represent two unique ASes. A provider authorization function, authorized(AS x, AS y), checks if the ordered pair of ASNs, (AS x, AS y), has the property that AS y is an attested provider of AS x per U-SPAS of AS x. By the term "Provider+", the function signals that AS y plays the role of Provider or non-transparent RS. This function is specified as follows:

```
                              /
                              | "No Attestation" if there is no entry
                              |   in U-SPAS table for CAS = AS x
                              |
   authorized(AS x, AS y) =   / Else, "Provider+" if the U-SPAS entry
                              \   for CAS = AS x includes AS y
                              |
                              | Else, "Not Provider+"
                              \
```

Figure 1: Provider authorization function.

The "No Attestation" result is returned only when no ASPA is retrieved for the CAS or none of its ASPAs are cryptographically valid. The provider authorization function is used in the ASPA-based AS_PATH verification algorithms described in Section 6.2 and Section 6.3.

6. AS_PATH Verification

The procedures described in this document are applicable only to four-octet AS number compatible BGP speakers [RFC6793]. If such a BGP speaker receives both AS_PATH and AS4_PATH attributes in an UPDATE, then the procedures are applied on the reconstructed AS_PATH (Section 4.2.3 of [RFC6793]). So, the term AS_PATH is used in this document to refer to the usual AS_PATH [RFC4271] as well as the reconstructed AS_PATH.

If an attacker creates a route leak intentionally, they may try to strip their AS from the AS_PATH. To partly guard against that, a check is necessary to match the most recently added AS in the AS_PATH to the BGP neighbor’s ASN. This check SHOULD be performed as specified in Section 6.3 of [RFC4271] with the exception when a route is received from transparent IX. If the check fails, then the AS_PATH is considered a Malformed AS_PATH and the UPDATE SHALL be handled using the approach of "treat-as-withdraw" [RFC7606].

[I-D.ietf-idr-deprecate-as-set-confed-set] specifies that "treat-as-withdraw" error handling [RFC7606] SHOULD be applied to routes with AS_SET in the AS_PATH. In the current document, routes with AS_SET are given Invalid evaluation in the AS_PATH verification procedures (Section 6.2 and Section 6.3). See Section 6.4 for how routes with Invalid AS_PATH are handled.

6.1. Principles

Let the sequence {AS(N), AS(N-1),..., AS(2), AS(1)} represent the AS_PATH in terms of unique ASNs, where AS(1) is the origin AS and AS(N) is the most recently added AS and neighbor of the receiving/verifying AS. N is the length of the received AS_PATH in unique ASes. AS(N+1) represents the local (receiving/verifying) AS; it does not explicitly appear in the description of the AS_PATH verification procedures.

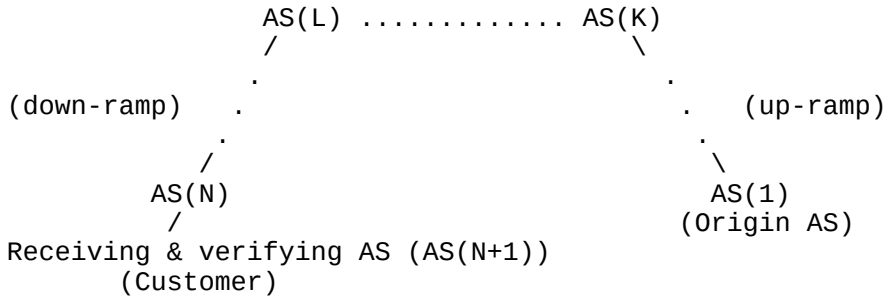

Figure 2: Illustration of up-ramp and down-ramp.

The AS_PATH may in general have both an up-ramp (on the right starting at AS(1)) and a down-ramp (on the left starting at AS(N)). The up-ramp runs from AS(1) to AS(K). In this ramp, each hop AS(i) to AS(i+1) represents a customer-to-provider peering relationship. The down-ramp runs backward from AS(N) to AS(L). In the down-ramp, each pair AS(j) to AS(j-1) represents a customer-to-provider peering relationship.

In the valid AS_PATH the apex AS(L) and AS(K) may be two different ASes or one AS. If they are the same AS, the up-ramp and down-ramp will have an overlap and the combined length of the up-ramp and down-ramp (measured in the number of ASes) will be equal to N+1. If AS(L) and AS(K) are different ASes, the combined length of the up-ramp and down-ramp will be equal to N.

Equivalently, if the combined length of the up-ramp and down-ramp is less than N the prefix was leaked or the AS_PATH was malformed.

ASPA can be used to check if AS y is an attested provider of AS x, and thus the provider authorization function can be used to measure the bounds on up-ramp and down-ramp lengths. The "Not Provider+" outcome of the provider authorization function can be used to calculate the upper boundary of ramp length and the 'No Attestation' outcome can be used to calculate its lower boundary. Below are the formal definitions.

Determine the maximum up-ramp length as I, where I is the minimum index for which authorized(A(I), A(I+1)) returns "Not Provider+". If there is no such I, the maximum up-ramp length is set equal to the AS_PATH length N. This parameter is abbreviated as max_up_ramp. The minimum up-ramp length can be determined as I, where I is the minimum index for which authorized(A(I), A(I+1)) returns "No Attestation" or "Not Provider+". If there is no such I, the AS_PATH consists of only "Provider+" pairs; so the minimum up-ramp length is set equal to the AS_PATH length N. This parameter is abbreviated as min_up_ramp.

Similarly, the maximum down-ramp length can be determined as N - J + 1 where J is the maximum index for which authorized(A(J), A(J-1)) returns "Not Provider+". If there is no such J, the maximum down-ramp length is set equal to the AS_PATH length N. This parameter is abbreviated as max_down_ramp. The minimum down-ramp length can be determined as N - J + 1 where J is the maximum index for which authorized(A(J), A(J-1)) returns "No Attestation" or "Not Provider+". If there is no such J, the minimum down-ramp length is set equal to the AS_PATH length N. This parameter is abbreviated as min_down_ramp.

If the sum of max_up_ramp and max_down_ramp is less than N, the AS_PATH is Invalid. Else, if the sum of min_up_ramp and min_down_ramp is less than N, enough information is not available to perform full AS_PATH verification, and the outcome is set to Unknown. Else, the AS_PATH is Valid.

Below are formal procedures for path verification depending on the peering relationship between the receiving AS and its neighbor. These procedures use the compressed sequence representation of AS_PATH {AS(N), AS(N-1),..., AS(2), AS(1)} and the above-defined parameters max_up_ramp, min_up_ramp, max_down_ramp, and min_up_ramp

6.2. Algorithm for Upstream Paths

The upstream verification algorithm described here is applied when a route is received from a Customer or Peer, or is received by an RS from an RS-client, or is received by an RS-client from an RS. In all these cases, the receiving/validating eBGP router expects the AS_PATH to have only an up-ramp (no down-ramp) for it to be Valid. Therefore, max_down_ramp and min_down_ramp are set to 0.

The upstream path verification procedure is specified as follows:

1. If the AS_PATH is empty, then the procedure halts with the outcome "Invalid".

2. If the receiving AS is not an RS-client and the most recently added AS in the AS_PATH does not match the neighbor AS, then the procedure halts with the outcome "Invalid".

3. If the AS_PATH has an AS_SET, then the procedure halts with the outcome "Invalid".

4. If max_up_ramp < N, the procedure halts with the outcome "Invalid".

5. If min_up_ramp < N, the procedure halts with the outcome "Unknown".

6. Else, the procedure halts with the outcome "Valid".

6.3. Algorithm for Downstream Paths

The downstream verification algorithm described here is applied when a route is received from a Provider.

1. If the AS_PATH is empty, then the procedure halts with the outcome "Invalid".

2. If the most recently added AS in the AS_PATH does not match the neighbor AS, then the procedure halts with the outcome "Invalid".

3. If the AS_PATH has an AS_SET, then the procedure halts with the outcome "Invalid".

4. If max_up_ramp + max_down_ramp < N, the procedure halts with the outcome "Invalid".

5. If min_up_ramp + min_down_ramp < N, the procedure halts with the outcome "Unknown".

6. Else, the procedure halts with the outcome "Valid".

6.4. Mitigation Policy

The specific configuration of a mitigation policy based on AS_PATH verification using ASPA is at the discretion of the network operator. However, the following mitigation policy is RECOMMENDED.

Invalid: If the AS_PATH is determined to be Invalid, then the route SHOULD be considered ineligible for route selection (see Section 3) and MUST be kept in the Adj-RIB-In for potential future re-evaluation (see [RFC9324]).

Valid or Unknown: When a route is evaluated as Unknown (using ASPA-based AS_PATH verification), it SHOULD be treated at the same preference level as a route evaluated as Valid.

7. Deployment Recommendations

This section describes practical deployment recommendations for applying verification procedures.

7.1. ASPA Verification Examples

A set of examples of AS_PATH verification using the above procedures (Section 6.1, Section 6.2, and Section 6.3) for an illustrative network topology are provided online (see [aspa-examples]).

7.2. Application of Verification Procedures

The verification procedures described in this document MUST be applied to BGP routes with {AFI, SAFI} combinations {AFI 1 (IPv4), SAFI 1} and {AFI 2 (IPv6), SAFI 1} [IANA-AF] [IANA-SAF]. The procedures MUST NOT be applied to other address families by default.

The procedures for ASPA-based AS_PATH verification are intended for implementation on edge routers on the ingress side. This includes edge routers on the boundary of an AS Confederation facing external ASes. However, the procedures are NOT RECOMMENDED for use on internal BGP (iBGP) sessions or eBGP sessions internal to an AS Confederation.

7.3. BGP Roles

The BGP Role configuration parameter and its cross-check in BGP OPEN message as specified in [RFC9234] are RECOMMENDED. The configured BGP Roles SHOULD be used to automate the use of the above-described AS_PATH verification procedures helping to distinguish whether upstream or downstream procedures should be applied. The automatic BGP Role cross-check [RFC9234] should facilitate more accurate and effective deployment of ASPA.

7.4. Complex Peering Relationships

If multiple eBGP sessions can segregate the Complex peering relationship into eBGP sessions with normal peering relationships the receiving/verifying AS SHOULD select the algorithm (per Section 6.2 or Section 6.3) for each of the normal sessions based on its peering relation type.

If a Complex peering relation cannot be segregated (i.e., when a Complex BGP relationship occurs within one single BGP session), an operator may want to achieve an equivalent outcome by applying an appropriate algorithm (Section 6.2 or Section 6.3) on a per-prefix basis corresponding to the peering relation for the prefix. If this option is not feasible, then an operator MAY apply the algorithm for downstream paths (Section 6.3) to avoid false positive outcomes.

7.5. Logging

For any route with an Invalid AS_PATH, the cause of the Invalid state SHOULD be logged for monitoring and diagnostic purposes. The cause of the Invalid state can be recorded in the form of listing the AS hops which were evaluated by the provider authorization function to be "Not Provider+". The logging router, however, cannot necessarily determine the AS that caused the route leak.

8. Security Considerations

8.1. Incongruence in IPv4 and IPv6 Connectivity

The U-SPAS contains the union of Providers for a CAS for both IPv4 and IPv6 unicast connectivity. This design choice consequently means that if a customer-provider relationship exists for one address family but doesn’t exist for the other address family, AS_PATH verification outcomes for the latter AFI will be as permissive as verification outcomes for the former AFI. That is believed to be a reasonable compromise as both the ASPA registration and verification processes are simplified, and no false positive outcomes are yielded (e.g. inadvertent Invalid evaluation).

8.2. Correctness of the ASPA

Network operators must keep their ASPA objects correct and up to date (Section 4). An incorrect or outdated ASPA may affect route propagation or limit route leak detection capabilities.

An AS SHOULD periodically monitor all ASPAs in global RPKI repositories to check if their AS number is included in the SPAS of their Customer ASes.

8.3. Manipulation of AS_PATH by Provider

A Provider may hijack prefixes of its direct or indirect customers by using forged-origin/forged-segment AS_PATH. It may also manipulate the AS_PATH of routes that are sent to its customers. Such attacks may not be detectable with ASPA.

While such attacks may happen in theory, it does not seem to be a realistic scenario. Normally a customer and their transit provider would have a signed agreement, and a policy violation (of the above kind) should have legal consequences or the customer can just drop the relationship with such a provider and remove the corresponding ASPA record.

8.4. Manipulating AS_PATH Prepends

The ASPA verification procedures cannot detect the removal (or addition) of repeats of AS numbers in the AS_PATH. However, this attack by itself does not affect ASPA’s route leak detection capability.

9. Relation to Other Technologies

9.1. ROA

A ROA [RFC6482] is a digitally signed object that binds an IP prefix to an AS number and is signed by the prefix holder. The RPKI-ROV procedure [RFC6483] [RFC6811] uses ROAs to verify that an AS is authorized to originate a specific prefix. The joint use of ROA and ASPA records and their corresponding verification procedures can establish a security trusted chain capable of detecting not only accidental route leaks but also malicious AS_PATH manipulations [bgp-cycling].

9.2. BGPsec

The BGPsec [RFC8205] protocol was designed to solve the problem of AS_PATH verification by including cryptographic signatures in BGP Update messages. It offers protection against unauthorized path modifications and assures that the BGPsec Update traveled the path shown in the BGPsec_PATH Attribute. However, it does not detect route leaks (valley-free violations). Thus, BGPsec and ASPA are complementary technologies.

9.3. Peerlock

The Peerlock mechanism [Peerlock] [Flexsealing] has a similar objective as the ASPA-based route leak protection mechanism described in this document. It is commonly deployed by large Internet carriers to protect each other from route leaks. Peerlock depends on a laborious manual process in which operators coordinate the distribution of unstructured Provider Authorizations through out-of-band means in a many-to-many fashion. On the other hand, ASPA’s use of the RPKI allows for automated, scalable, and ubiquitous deployment, making the protection mechanism available to a wider range of network operators.

The ASPA mechanism implemented in router code (in contrast to Peerlock’s AS_PATH regular expressions) also provides a way to detect anomalies propagated from transit providers and IX route servers. ASPA is intended to be a complete solution and replacement for existing Peerlock deployments.

9.4. Only to Customer (OTC) Attribute

While the ASPA-based AS_PATH verification method (Section 6, Section 6.4) detects and mitigates route leaks that were created by preceding ASes listed in the AS_PATH, it lacks the ability to prevent the local AS from initiating a route leak towards its neighbor. ASPA verification may also fail to detect route leaks in case of presence of Complex relations in the AS_PATH. The use of the Only to Customer (OTC) Attribute fills in that gap (see Section 5, [RFC9234]). The implementation of the procedures utilizing the OTC Attribute is RECOMMENDED to complement the ASPA-based AS_PATH verification.

10. IANA Considerations

This document includes no request to IANA.

11. Implementation Status

This section is to be removed before publishing as an RFC.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft. The inclusion of this section here follows the process described in [RFC7942]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC7942], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

* A BGP implementation OpenBGPD [bgpd] (version 7.8 and higher), written in C, was provided by Claudio Jeker, Theo Buehler, and Job Snijders.

* The implementation NIST-BGP-SRx [BGP-SRx] is a software suite that provides a validation engine (BGP-SRx) and a Quagga-based BGP router (Quagga-SRx). It includes unit test cases for testing the ASPA-based path verification. It was provided by Oliver Borchert, Kyehwan Lee, and their colleagues at US NIST. It requires some additional work to incorporate the latest changes in the draft specifications related to IXP RS AS and RS-client.

* Implementation of ASPA-based AS_PATH verification in the BIRD Internet Routing Daemon [BIRD] is provided in a side branch (branch mq-aspa) by Katerina Kubecova and Maria Matejka. Its release is expected after RTR v2 is finalized.

* Implementation of ASPA-based AS_PATH verification in the FreeRTR [FreeRTR] is provided by Csaba Mate. Csaba reports that this implementation has passed all the unit tests listed in [aspa-examples].

12. References

12.1. Normative References

[I-D.ietf-sidrops-aspa-profile]
Azimov, A., Uskov, E., Bush, R., Snijders, J., Housley, R., and B. Maddison, "A Profile for Autonomous System Provider Authorization", Work in Progress, Internet-Draft, draft-ietf-sidrops-aspa-profile-19, 6 January 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-aspa-profile-19>.

[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>.

[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>.

[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>.

[RFC6793] Vohra, Q. and E. Chen, "BGP Support for Four-Octet Autonomous System (AS) Number Space", RFC 6793, DOI 10.17487/RFC6793, December 2012, <https://www.rfc-editor.org/info/rfc6793>.

[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>.

[RFC7606] Chen, E., Ed., Scudder, J., Ed., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", RFC 7606, DOI 10.17487/RFC7606, August 2015, <https://www.rfc-editor.org/info/rfc7606>.

[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>.

[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>.

[RFC9234] Azimov, A., Bogomazov, E., Bush, R., Patel, K., and K. Sriram, "Route Leak Prevention and Detection Using Roles in UPDATE and OPEN Messages", RFC 9234, DOI 10.17487/RFC9234, May 2022, <https://www.rfc-editor.org/info/rfc9234>.

[RFC9324] Bush, R., Patel, K., Smith, P., and M. Tinka, "Policy Based on the Resource Public Key Infrastructure (RPKI) without Route Refresh", RFC 9324, DOI 10.17487/RFC9324, December 2022, <https://www.rfc-editor.org/info/rfc9324>.

12.2. Informative References

[aspa-examples] "ASPA-based AS Path Verification Examples", <https://github.com/ksriram25/IETF/blob/main/ASPA_path_verification_examples.pdf>.

[bgp-cycling] Azimov, A., "BGP Route Security Cycling to the Future!", NANOG-76, North American Network Operator Group Meeting, Slides archives from NANOG, October 2019, <https://pc.nanog.org/static/published/meetings/NANOG76/1978/20190611_Azimov_Bgp_Route_Security_v1.pdf>.

[BGP-SRx] Lee, K. and O. Borchert, et al., "BGP Secure Routing Extension (BGP-SRx) Software Suite", NIST Open-Source Software , <https://www.nist.gov/services-resources/software/bgp-secure-routing-extension-bgp-srx-software-suite>.

[bgpd] Jeker, C., "OpenBGPD", <http://www.openbgpd.org/>.

[BIRD] Kubecova, K. and M. Matejka, "BIRD Internet Routing Daemon; branch mq-aspa", CZ.NIC BIRD Open-Source Software , <https://bird.nic.cz/en/>.

[Flexsealing] McDaniel, T., Smith, J., and M. Schuchard, "Flexsealing BGP Against Route Leaks: Peerlock Active Measurement and Analysis", November 2020, <https://arxiv.org/pdf/2006.06576.pdf>.

[FreeRTR] Mate, C., "FreeRTR", <http://www.freertr.org/>.

[I-D.ietf-grow-route-leak-detection-mitigation]
Sriram, K. and A. Azimov, "Methods for Detection and Mitigation of BGP Route Leaks", Work in Progress, Internet-Draft, draft-ietf-grow-route-leak-detection-mitigation-12, 25 February 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-grow-route-leak-detection-mitigation-12>.

[I-D.ietf-idr-deprecate-as-set-confed-set]
Kumari, W., Sriram, K., Hannachi, L., and J. Haas, "Deprecation of AS_SET and AS_CONFED_SET in BGP", Work in Progress, Internet-Draft, draft-ietf-idr-deprecate-as-set-confed-set-18, 7 March 2025, <https://datatracker.ietf.org/api/v1/doc/document/draft-ietf-idr-deprecate-as-set-confed-set/>.

[IANA-AF] IANA, "Address Family Numbers", <https://www.iana.org/assignments/address-family-numbers/address-family-numbers.xhtml>.

[IANA-SAF] IANA, "Subsequent Address Family Identifiers (SAFI) Parameters", <https://www.iana.org/assignments/safi-namespace/safi-namespace.xhtml>.

[nanog-aspa]
Sriram, K., "ASPA-based BGP AS_PATH Verification and Route Leaks Solution", NANOG-89, North American Network Operator Group Meeting, Slides/video archives from NANOG, October 2023, <https://storage.googleapis.com/site-media-prod/meetings/NANOG89/4809/20231017_Sriram_Aspa-Based_Bgp_As_Path_v1.pdf (slides) https://www.youtube.com/watch?v=GdVnZGd7jMo (video)>.

[Peerlock] Snijders, J., "Peerlock", June 2016, <https://www.nanog.org/sites/default/files/Snijders_Everyday_Practical_Bgp.pdf>.

[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>.

[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>.

[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>.

[RFC9319] Gilad, Y., Goldberg, S., Sriram, K., Snijders, J., and B. Maddison, "The Use of maxLength in the Resource Public Key Infrastructure (RPKI)", BCP 185, RFC 9319, DOI 10.17487/RFC9319, October 2022, <https://www.rfc-editor.org/info/rfc9319>.

Appendix A. Acknowledgments

The authors wish to thank Claudio Jeker, Jakob Heitz, Amir Herzberg, Igor Lubashev, Ben Maddison, Russ Housley, Jeff Haas, Nan Geng, Mingqing Huang, Nick Hilliard, Shunwan Zhuang, Yangyang Wang, Martin Hoffmann, Jay Borkenhagen, Amreesh Phokeer, Aftab Siddiqui, Dai Zhibin, Doug Montgomery, Padma Krishnaswamy, Rich Compton, Andrei Robachevsky, Rudiger Volk, Iljitsch van Beijnum, Tassilo Tanneberger, Matthias Waehlisch, Moritz Schulz, and Carl Seifert for comments, suggestions, and discussion on the path verification procedures or the text in the document. For the implementation and testing of the procedures in the document, the authors wish to thank Claudio Jeker and Theo Buehler [bgpd], Kyehwan Lee and Oliver Borchert [BGP-SRx], Katerina Kubecova and Maria Matejka [BIRD], and Csaba Mate [FreeRTR].

Appendix B. Properties and Early Adoption Benefits

The ASPA method has the properties (i.e., anomaly detection capabilities) listed below. Partial deployment scenarios and early adoption benefits are considered. In the case of Property 1, it is assumed that the attacks involve route leaks but not malicious removal of ASes with ASPA records from the AS_PATH.

Property 1 (Route Leak Detection): Let AS A and AS B be any two ASes in the Internet doing ASPA (registration and path verification) and no assumption is made about the ASPA deployment status of other ASes. Consider a route propagated from AS A to a customer or lateral peer. The route is subsequently leaked by an offending AS in the AS path before being received at AS B on a customer or lateral peer interface. The ASPA-based path verification at AS B always detects such a route leak though it may not be able to identify the AS that caused the leak.

Corollary of Property 1: An observation that follows from Property #1 above is that if any two ISP ASes register ASPAs and implement the detection and mitigation procedures, then any route received from one of them and leaked to the other by an AS in their overlapping customer cones (ASPA compliant or not) will be

automatically detected and mitigated. In effect, if most major ISPs are compliant, the propagation of route leaks in the Internet will be severely limited.

Property 2 (Detection of Forged-Origin Prefix Hijack): Again, let AS A and AS B be any two ASes in the Internet doing ASPA (registration and path verification) and no assumption is made about the ASPA deployment status of other ASes. Consider a route received at AS B on a customer or lateral peer interface that is a forged-origin prefix hijack involving AS A as the forged-origin. Assume that the offending AS X is not included in the ASPA of AS A. The ASPA-based path verification at AS B always detects such a forged-origin prefix hijack.

Property 3 (Detection of Forged-Path-Segment Prefix Hijack): This is an extension of Property 2 above to the case of prefix hijacking with a forged-path-segment. Such hijacking refers to the forging of multiple contiguous ASes in an AS path beginning with the origin AS. Again, let AS A and AS B be any two ASes in the Internet doing ASPA (registration and path verification). Assume that AS A's providers, AS P and AS Q, also register ASPA. No assumption is made about the ASPA deployment status of any other ASes in the Internet. Consider a route received at AS B on a customer or lateral peer interface that is a prefix hijack with a forged-path-segment {AS P, AS A} or {AS Q, AS A}. That is, the offending AS X attaches this path-segment at the beginning of its (AS X's) route announcement. Assume that AS X is not included in the ASPA of AS P or AS Q. The ASPA-based path verification at AS B always detects such a forged-path-segment prefix hijack. For a chance to be successful (remain undetected by AS B), the hijacker may resort to a forged-path-segment with three ASes including a provider AS of AS P (or AS Q). But even that can be foiled (detected) if the providers of AS P and AS Q also register ASPA. The forged-path-segment hijack in consideration is entirely prevented (for any forged-path-segment length) if all ASes in the contiguous customer-to-provider (C2P) hops from AS A up to and including the topmost tier AS have ASPA registrations.

The above properties show that ASPA-based path verification offers significant benefits to early adopters (also see [nanog-aspa]). Limitations of the method with regard to some forms of malicious AS path manipulations are discussed in Section 8.

Authors' Addresses

Alexander Azimov
Yandex
Ulitsa Lva Tolstogo 16

Moscow
119021
Russian Federation
Email: a.e.azimov@gmail.com

Eugene Bogomazov
Qrator Labs
1-y Magistralnyy tupik 5A
Moscow
123290
Russian Federation
Email: eb@qrator.net

Randy Bush
Internet Initiative Japan & Arrcus, Inc.
5147 Crystal Springs
Bainbridge Island, Washington 98110
United States of America
Email: randy@psg.com

Keyur Patel
Arrcus
2077 Gateway Place
Suite #400
San Jose, CA 95119
United States of America
Email: keyur@arrcus.com

Job Snijders
Amsterdam
Netherlands
Email: job@sobornost.net

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