Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 4 September 2024

H. Chen
China Telecom
D. Ma
ZDNS
N. Geng
S. Zhuang
H. Wang
Huawei

3 March 2024

Enhanced AS-Loop Detection for BGP
draft-chen-grow-enhanced-as-loop-detection-07

Abstract

Misconfiguration and malicious manipulation of BGP AS_Path may lead to route hijack. This document proposes to enhance the BGP [RFC4271] Inbound/ Outbound route processing in the case of detecting an AS loop. It is an enhancement to the current BGP's Inbound/Outbound processing and can be implemented directly on the device, and this document also proposes a centralized usecase. This could empower networks to quickly and accurately figure out they're being victimized.

Two options are proposed for the enhancement, a) a local check at the device; b) data collection/analysis at the remote network controller/ server. Both approaches are beneficial for route hijack detection.

Requirements Language

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 RFC 2119 [RFC2119].

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The Border Gateway Protocol (BGP) [RFC4271], as an inter-autonomous (AS) routing protocol, is used to exchange network reachability information between BGP systems. BGP is widely used by Internet Service Providers (ISPs) and large organizations.

As a distance-vector based protocol, BGP is used to exchange reachable inter-AS routes, establish inter-AS paths, avoid routing loops, and apply routing policies between ASs. BGP loop detection mechanism is defined in section 9.1.2. of RFC4271:

> ...
>
> If the AS_PATH attribute of a BGP route contains an AS loop, the BGP route should be excluded from the Phase 2 decision function. AS loop detection is done by scanning the full AS path (as specified in the AS_PATH attribute), and checking that the autonomous system number of the local system does not appear in the AS path. Operations of a BGP speaker that is configured to accept routes with its own autonomous system number in the AS path are outside the scope of this document.
>
> ...

In ordinary BGP, every AS announces its route information with different prefixes. However, its neighboring ASes cannot validate this route information, but rather directly propagate it across the Internet or simply discard AS-Loop routes directly. Obviously, this weak trust model allows forged route announcement propagations and rarely been found, which is a fundamental security weakness of BGP. Forged routes, which can be generated by configuration errors or malicious attacks, can lead to large-scale network connectivity issues.

Some cases can be worse, hackers exploit this property of BGP to achieve their ulterior motives. They can add some providers' AS number into the forged AS-Path and attempt to make it look like the route had passed through these ASNs, or perhaps they are there to prevent those providers from carrying the route. These cases are also being known As-Path Poisoning Attacks.

ASPA [I-D.ietf-sidrops-aspa-verification] can be used to verify the AS_PATH attribute of routes advertised in the Border Gateway Protocol, and it is a systematic deployment based on RPKI system. This mechanism requires a series of infrastructure implementations.

This document proposes to enhance AS-Loop Detection for BGP Inbound/ Outbound Route Processing when detecting AS loop in order to identify possible BGP hijacks. It is an enhancement to the current BGP's Inbound/Outbound processing and can be implemented directly on the device, and this document also proposes a centralized usecase. This could empower networks to quickly and accurately figure out they're being victimized.

2. Terminology

The following terminology is used in this document.

AS: Autonomous System

ASPA: Autonomous System Provider Authorization

BGP: Border Gateway Protocol

BGP hijacking : is the illegitimate takeover of groups of IP addresses by corrupting Internet routing tables maintained using the Border Gateway Protocol (BGP). (Sometimes referred to as prefix hijacking, route hijacking or IP hijacking)

EBGP: External BGP

ISP: Internet Service Provider

BMP: BGP Monitoring Protocol

ROA: Route Origin Authorization

3. Forged AS_PATH Examples

3.1. AS Loop Detected at Inbound Processing

* Forged Case 1: AS shown in Figure 1, an upstream AS of AS64596 forged a route with the ASN 64596 as the origin ASN in the AS-Path.

* Forged Case 2: AS shown in Figure 1, an upstream AS of AS64596 forged a route with the ASN 64596 as the transit ASN in the AS-Path.

```
   AS-Loop-Detecting at this point
   Discard AS-Loop Routes directly that contains AS64596
                  |
                  |                                 x.y.z.0/24
                  v                                 Origin AS 64600
AS64595---AS64596---AS64597---AS64598---AS64599----AS64600
                    From AS64597, To AS64596:
                    Normal Case:
                    <-- x.y.z.0/24, AS-Path: 64597 64598 64599 64600

                    Forged Case 1:
                    <-- x.y.z.0/24, AS-Path: 64597 64596
                                       (Or: 64597 64598 64596 etc.)

                    Forged Case 2:
                    <-- x.y.z.0/24, AS-Path: 64597 64596 64600
                                       (Or: 64597 64596 64599 64600 etc.)
```

Figure 1: BGP Inbound Route Processing in AS64596

After receiving the above routes, AS64596 treats them as normal loop routes during the loop detecting phase and discards them directly. In most NOSes (Network Operation Systems), such rejected routes are not logged and only visible by putting the router into debugging mode. If the AS64596 is slightly enhanced, it can find that someone has faked himself, which may cause unnecessary trouble for himself.

3.2. AS Loop Detected at Outbound Processing

Split-Horizon for EBGP is an optional function that a BGP sender will not advertise any routes that were previously received from that same AS. In some current implementation, the BGP outbound route processing step will simply discard the route if AS-Loop being detected.

* Forged Case 3: AS shown in Figure 2, an upstream AS of AS64597 forged a route with the ASN 64596 as the origin ASN in the AS-Path.

* Forged Case 4: AS shown in Figure 2, an upstream AS of AS64597 forged a route with the ASN 64596 as the transit ASN in the AS-Path.

```
    Split-Horizon Enable & AS-Loop-Detecting at this point
    Discard AS-Loop Routes directly if sending AS-Path contains AS64596
                    |
                    |                               x.y.z.0/24
                    v                               Origin AS 64600
AS64595---AS64596---AS64597---AS64598---AS64599----AS64600
                     From AS64597, To AS64596:
                     Normal Case:
                     <-- x.y.z.0/24, AS-Path: 64597 64598 64599 64600

                     Forged Case 3:
                     <-- x.y.z.0/24, AS-Path: 64597 64596
                                            (Or: 64597 64598 64596 etc.)

                     Forged Case 4:
                     <-- x.y.z.0/24, AS-Path: 64597 64596 64600
                                        (Or: 64597 64596 64599 64600 etc.)
```

Figure 2: BGP Outbound Route Processing in AS64597

When sending the above routes, AS64597 treats them as normal loop routes and discards them directly. If AS64597 is slightly enhanced, it can find that someone has faked AS64596, which may cause large-scale network connectivity problems.

4. Enhancement to BGP Inbound/Outbound Processing

4.1. Enhancement for AS Loop Detected at Inbound Process

Currently, ROV [RFC6811] and ASPA verification [I-D.ietf-sidrops-aspa-verification] can be adopted for BGP leak/hijack detection. However, for the forged case 1&2, the conventional BGP inbound process would simply discard the routes with AS loop before any further leak/hajack detection.

This document suggests further analysis of such routes. The analysis may include mechanisms that apply to normal routes for hijack detection, such as ROV, ASPA and so on. The detailed analyzing mechanisms as well as the corresponding actions w.r.t. the analysis are outside the scope of this document. Two options of where the analysis of the inbound processing enhancement takes place is proposed.

* Option 1: Analyze the routes with AS loop based on local database.

* Option 2: Collect the routes with AS loop with BMP and analyze them at the remote controller/server.

4.2. Enhancement for AS Loop Detected at Outbound Process

Currently, the egress ROV can be adopted for BGP hijack detection. However, for forged case 3&4, when eBGP Split-Horizon is enabled, the routes with AS loop could possibly be discarded before any hijack detection.

This document suggests further analysis of such routes. The analysis may include mechanisms that apply to normal routes for hijack detection, such as egress ROV, ASPA and so on. The detailed analyzing mechanisms as well as the corresponding actions w.r.t. the analysis are outside the scope of this document.

Two options of where the analysis of the outbound processing enhancement takes place is proposed.

* Option 1: Analyze the routes with AS loop based on local database.

* Option 2: Collect the routes with AS loop with BMP and analyze them at the remote controller/server.

5. Centralized AS-Loop Detection for BGP

Considering the challenges facing the existing approaches, this section proposes a centralized method. It utilizes the BGP Monitoring Protocol (BMP) to convey the AS Path Looped Update message from the monitored device to the BMP server to realize centralized attack detection.

BMP is currently deployed by OTT and Carriers to monitor the BGP routes, such as monitoring BGP Adj-RIB-In using the process defined in RFC7854 [RFC7854], and monitoring BGP Adj-RIB-Out using the process defined in RFC8761 [RFC8671]. This document extends Route Mirroring message to mirror AS Path Looped update message to the BMP Server.

5.1. BMP Support for Monitoring AS Path Looped Update Message

Per RFC7854, Route Mirroring messages can be used to mirror the messages that have been treated-as-withdraw [RFC7606], for debugging purposes. This document extends Route Mirroring message to mirror AS Path Looped update message to the BMP Server.

This document adds a new code for Type 1 Information TLV:

* Code = TBD: AS Path Looped. The BGP Message TLV occurs in the Route Mirroring message and whose loop includes the local AS.

Following the common BMP header and per-peer header is an Information TLV (Type = 1) with Code = TBD: AS Path Looped, and then a BGP Message TLV (Type = 0) contain an AS Path Looped Update Message.

```
+-------------------------------------------------------------+
|         Common BMP Header (Message Type = 6)                |
+-------------------------------------------------------------+
|                    Per peer Header                          |
+-------------------------------------------------------------+
| Information TLV (Type = 1) with Code = TBD: AS Path Looped  |
+-------------------------------------------------------------+
|         BGP Message TLV (Type = 0)                          |
+-------------------------------------------------------------+
```

Figure 3: AS Path Looped Update Message Carrying in the Route Mirroring Message

5.2. Application Example

This section describe a centralized application example. As shown in Figure 4, when receiving the routes from AS64597, AS64596 should check whether its own AS number is already in the AS-Path, If yes, it further encapsulate the AS Path Looped Update Message in the Route Mirroring message and sends the Route Mirroring message to the BMP Server.

The Analyzer gets the AS Path Looped Update Messages from the BMP Server and further processes them.

```
                          +------------+
                          | BMP server |
                          |     +      |
                          |  Analyzer  |
                          +------+-----+
    BMP Mirroring Message:      \
    AS Path Looped Update Message\to BMP Server
                                  \
                                   \
                 ********************|****
                 *                   |   *
                 *          AS64596  |   *        "AS64597 Send Routes
                 *                   |   *  <----- to AS64596"
   +-------+     *  +---+         +---+  *  +-------+
   +AS64595+-------+ R1+---------+ R2|-----+AS64597+---64598 64599 64600
   +-------+     *  +-+-+\        +---+  *  +-------+
                 *    |   \\    //  |\   *
                 *    |     \\//    | \  *
                 *    |     //\     |  \ *      +-------+
                 *    |   //   \\   |   \+------+AS64593+
                 *    |  /       \  |    *      +-------+
   +-------+     *  +-+-+         +-+-+  *      +-------+
   +AS64594+----+--+ R3+---------+ R4+---------+AS64592+
   +-------+     *  +---+         +---+  *      +-------+
                 *                       *
                 ************************
```

Figure 4: Centralized AS-Loop Detection

From the perspective of the local AS, it can manage/hold the AS-relationship database between the local AS and each of its neighboring ASs (such as C2P, P2P, P2C, etc.).

Neighboring AS	AS-relationship to AS64596
64592	P2P
64593	S2S
64594	C2P
64595	P2C
64597	P2P

Figure 5: AS64596's AS-Relationship Database

When AS 64596 is listed as transit AS in the AS-Path, for example, AS-Path looks like the following form AS64596's perspective:

(possible other ASes), left AS, local AS(64596), right AS, (possible other ASes)

At this point, AS64596's Analyzer can lookup the local resource database and check whether there is a real AS relationship between the local AS and the left AS and the right AS.

6. Benefits

After the enhancements of the AS Loop Detection for BGP Inbound/ Outbound Route Processing are added, the stability and security of the network can be improved.

7. Acknowledgements

The authors would like to acknowledge the review and inputs from Gang Yan, Zhenbin Li, Aijun Wang, Jeff Haas, Robert Raszuk, Chris Morrow, Alexander Asimov, Ruediger Volk, Jescia Chen and the working group.

8. IANA Considerations

This document defines one type for information carried in the Route Mirroring Information (Section 4.7 of RFC7854) code:

* Code = TBD: AS Path Looped.

9. Security Considerations

This document does not change the underlying security issues in the BGP protocol. It however, does provide an additional mechanism to protect against attacks based on the forged AS-Path in the BGP routes.

10. Contributors

The following people made significant contributions to this document:

Yunan Gu

Huawei Email: guyunan@huawei.com

11. Normative References

[I-D.ietf-sidrops-aspa-verification]
Azimov, A., Bogomazov, E., Bush, R., Patel, K., Snijders, J., and K. Sriram, "BGP AS_PATH Verification Based on Autonomous System Provider Authorization (ASPA) Objects", Work in Progress, Internet-Draft, draft-ietf-sidrops-aspa-verification-17, 29 February 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-aspa-verification-17>.

[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>.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <https://www.rfc-editor.org/info/rfc4760>.

[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>.

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/info/rfc7854>.

[RFC8671] Evens, T., Bayraktar, S., Lucente, P., Mi, P., and S. Zhuang, "Support for Adj-RIB-Out in the BGP Monitoring Protocol (BMP)", RFC 8671, DOI 10.17487/RFC8671, November 2019, <https://www.rfc-editor.org/info/rfc8671>.

Authors' Addresses

Huanan Chen
China Telecom
109, West Zhongshan Road, Tianhe District
Guangzhou
510000
China
Email: chenhuan6@chinatelecom.cn

Di Ma
ZDNS
4 South 4th St. Zhongguancun
Beijing
Haidian,
China
Email: madi@zdns.cn

Nan Geng
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: gengnan@huawei.com

Shunwan Zhuang
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: zhuangshunwan@huawei.com

Haibo Wang
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: rainsword.wang@huawei.com

Global Routing Operations
Internet-Draft
Updates: 7854 (if approved)
Intended status: Standards Track
Expires: February 6, 2020

T. Evens
S. Bayraktar
Cisco Systems
P. Lucente
NTT Communications
P. Mi
Tencent
S. Zhuang
Huawei

August 5, 2019

Support for Adj-RIB-Out in BGP Monitoring Protocol (BMP)
draft-ietf-grow-bmp-adj-rib-out-07

Abstract

The BGP Monitoring Protocol (BMP) defines access to only the Adj-RIB-In Routing Information Bases (RIBs). This document updates the BGP Monitoring Protocol (BMP) RFC 7854 by adding access to the Adj-RIB-Out RIBs. It adds a new flag to the peer header to distinguish Adj-RIB-In and Adj-RIB-Out.

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 February 6, 2020.

Table of Contents

1. Introduction

BGP Monitoring Protocol (BMP) defines monitoring of the received (e.g., Adj-RIB-In) Routing Information Bases (RIBs) per peer. The Adj-RIB-In pre-policy conveys to a BMP receiver all RIB data before any policy has been applied. The Adj-RIB-In post-policy conveys to a BMP receiver all RIB data after policy filters and/or modifications have been applied. An example of pre-policy versus post-policy is when an inbound policy applies attribute modification or filters. Pre-policy would contain information prior to the inbound policy changes or filters of data. Post policy would convey the changed data or would not contain the filtered data.

Monitoring the received updates that the router received before any policy has been applied is the primary level of monitoring for most use-cases. Inbound policy validation and auditing is the primary use-case for enabling post-policy monitoring.

In order for a BMP receiver to receive any BGP data, the BMP sender (e.g., router) needs to have an established BGP peering session and actively be receiving updates for an Adj-RIB-In.

Being able to only monitor the Adj-RIB-In puts a restriction on what data is available to BMP receivers via BMP senders (e.g., routers). This is an issue when the receiving end of the BGP peer is not enabled for BMP or when it is not accessible for administrative reasons. For example, a service provider advertises prefixes to a customer, but the service provider cannot see what it advertises via BMP. Asking the customer to enable BMP and monitoring of the Adj-RIB-In is not feasible.

BGP Monitoring Protocol (BMP) RFC 7854 [RFC7854] only defines Adj-RIB-In being sent to BMP receivers. This document updates the per-peer header in section 4.2 of [RFC7854] by adding a new flag to distinguish Adj-RIB-In versus Adj-RIB-Out. BMP senders use the new flag to send either Adj-RIB-In or Adj-RIB-Out.

Adding Adj-RIB-Out provides the ability for a BMP sender to send to BMP receivers what it advertises to BGP peers, which can be used for outbound policy validation and to monitor routes that were advertised.

2. Terminology

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 RFC 2119 [RFC2119] RFC 8174 [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Definitions

- Adj-RIB-Out: As defined in [RFC4271], "The Adj-RIBs-Out contains the routes for advertisement to specific peers by means of the local speaker’s UPDATE messages."

- Pre-Policy Adj-RIB-Out: The result before applying the outbound policy to an Adj-RIB-Out. This normally would match what is in the local RIB.

- o Post-Policy Adj-RIB-Out: The result of applying outbound policy to an Adj-RIB-Out. This MUST convey to the BMP receiver what is actually transmitted to the peer.

4. Per-Peer Header

The per-peer header has the same structure and flags as defined in section 4.2 of [RFC7854] with the following O flag addition:

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|V|L|A|O| Resv  |
+-+-+-+-+-+-+-+-+
```

- o The O flag indicates Adj-RIB-In if set to 0 and Adj-RIB-Out if set to 1.

The existing flags are defined in section 4.2 of [RFC7854] and the remaining bits are reserved for future use. They MUST be transmitted as 0 and their values MUST be ignored on receipt.

When the O flag is set to 1, the following fields in the Per-Peer Header are redefined:

- o Peer Address: The remote IP address associated with the TCP session over which the encapsulated PDU is sent.

- o Peer AS: The Autonomous System number of the peer to which the encapsulated PDU is sent.

- o Peer BGP ID: The BGP Identifier of the peer to which the encapsulated PDU is sent.

- o Timestamp: The time when the encapsulated routes were advertised (one may also think of this as the time when they were installed in the Adj-RIB-Out), expressed in seconds and microseconds since midnight (zero hour), January 1, 1970 (UTC). If zero, the time is unavailable. Precision of the timestamp is implementation-dependent.

5. Adj-RIB-Out

5.1. Post-Policy

The primary use-case in monitoring Adj-RIB-Out is to monitor the updates transmitted to a BGP peer after outbound policy has been applied. These updates reflect the result after modifications and filters have been applied (e.g., Adj-RIB-Out Post-Policy). Some

attributes are set when the BGP message is transmitted, such as next-hop. Adj-RIB-Out Post-Policy MUST convey to the BMP receiver what is actually transmitted to the peer.

The L flag MUST be set to 1 to indicate post-policy.

5.2. Pre-Policy

Similarly to Adj-RIB-In policy validation, pre-policy Adj-RIB-Out can be used to validate and audit outbound policies. For example, a comparison between pre-policy and post-policy can be used to validate the outbound policy.

Depending on BGP peering session type (IBGP, IBGP route reflector client, EBGP, BGP confederations, Route Server Client) the candidate routes that make up the Pre-Policy Adj-RIB-Out do not contain all local-rib routes. Pre-Policy Adj-RIB-Out conveys only routes that are available based on the peering type. Post-Policy represents the filtered/changed routes from the available routes.

Some attributes are set only during transmission of the BGP message, i.e., Post-Policy. It is common that next-hop may be null, loopback, or similar during pre-policy phase. All mandatory attributes, such as next-hop, MUST be either ZERO or have an empty length if they are unknown at the Pre-Policy phase completion. The BMP receiver will treat zero or empty mandatory attributes as self-originated.

The L flag MUST be set to 0 to indicate pre-policy.

6. BMP Messages

Many BMP messages have a per-peer header but some are not applicable to Adj-RIB-In or Adj-RIB-Out monitoring, such as peer up and down notifications. Unless otherwise defined, the O flag should be set to 0 in the per-peer header in BMP messages.

6.1. Route Monitoring and Route Mirroring

The O flag MUST be set accordingly to indicate if the route monitor or route mirroring message conveys Adj-RIB-In or Adj-RIB-Out.

6.2. Statistics Report

The Statistics report message has a Stat Type field to indicate the statistic carried in the Stat Data field. Statistics report messages are not specific to Adj-RIB-In or Adj-RIB-Out and MUST have the O flag set to zero. The O flag SHOULD be ignored by the BMP receiver.

The following new statistic types are added:

- o Stat Type = 14: (64-bit Gauge) Number of routes in Adj-RIBs-Out Pre-Policy.

- o Stat Type = 15: (64-bit Gauge) Number of routes in Adj-RIBs-Out Post-Policy.

- o Stat Type = 16: Number of routes in per-AFI/SAFI Adj-RIB-Out Pre-Policy. The value is structured as: 2-byte Address Family Identifier (AFI), 1-byte Subsequent Address Family Identifier (SAFI), followed by a 64-bit Gauge.

- o Stat Type = 17: Number of routes in per-AFI/SAFI Adj-RIB-Out Post-Policy. The value is structured as: 2-byte Address Family Identifier (AFI), 1-byte Subsequent Address Family Identifier (SAFI), followed by a 64-bit Gauge.

6.3. Peer Down and Up Notifications

Peer Up and Down notifications convey BGP peering session state to BMP receivers. The state is independent of whether or not route monitoring or route mirroring messages will be sent for Adj-RIB-In, Adj-RIB-Out, or both. BMP receiver implementations SHOULD ignore the O flag in Peer Up and Down notifications.

6.3.1. Peer Up Information

The following Peer Up message Information TLV type is added:

- o Type = 4: Admin Label. The Information field contains a free-form UTF-8 string whose byte length is given by the Information Length field. The value is administratively assigned. There is no requirement to terminate the string with null or any other character.

  Multiple admin labels can be included in the Peer Up notification. When multiple admin labels are included the BMP receiver MUST preserve their order.

  The TLV is optional.

7. Other Considerations

7.1. Peer and Update Groups

Peer and update groups are used to group updates shared by many peers. This is a level of efficiency in implementations, not a true representation of what is conveyed to a peer in either Pre-Policy or Post-Policy.

One of the use-cases to monitor Adj-RIB-Out Post-Policy is to validate and continually ensure the egress updates match what is expected. For example, wholesale peers should never have routes with community X:Y sent to them. In this use-case, there may be hundreds of wholesale peers but a single peer could have represented the group.

From a BMP perspective, this should be simple to include a group name in the Peer Up, but it is more complex than that. BGP implementations have evolved to provide comprehensive and structured policy grouping, such as session, AFI/SAFI, and template-based based group policy inheritances.

This level of structure and inheritance of polices does not provide a simple peer group name or ID, such as wholesale peer.

Instead of requiring a group name to be used, a new administrative label informational TLV (Section 6.3.1) is added to the Peer Up message. These labels have administrative scope relevance. For example, labels "type=wholesale" and "region=west" could be used to monitor expected policies.

Configuration and assignment of labels to peers is BGP implementation specific.

8. Security Considerations

The same considerations as in section 11 of [RFC7854] apply to this document. Implementations of this protocol SHOULD require to establish sessions with authorized and trusted monitoring devices. It is also believed that this document does not add any additional security considerations.

9. IANA Considerations

This document requests that IANA assign the following new parameters to the BMP parameters name space [1].

9.1. BMP Peer Flags

This document defines the following per-peer header flags (Section 4):

- o Flag 3 as O flag: The O flag indicates Adj-RIB-In if set to 0 and Adj-RIB-Out if set to 1.

9.2. BMP Statistics Types

This document defines four statistic types for statistics reporting (Section 6.2):

- o Stat Type = 14: (64-bit Gauge) Number of routes in Adj-RIBs-Out Pre-Policy.

- o Stat Type = 15: (64-bit Gauge) Number of routes in Adj-RIBs-Out Post-Policy.

- o Stat Type = 16: Number of routes in per-AFI/SAFI Adj-RIB-Out Pre-Policy. The value is structured as: 2-byte Address Family Identifier (AFI), 1-byte Subsequent Address Family Identifier (SAFI), followed by a 64-bit Gauge.

- o Stat Type = 17: Number of routes in per-AFI/SAFI Adj-RIB-Out Post-Policy. The value is structured as: 2-byte Address Family Identifier (AFI), 1-byte Subsequent Address Family Identifier (SAFI), followed by a 64-bit Gauge.

9.3. Peer Up Information TLV

This document defines the following BMP Peer Up Information TLV types (Section 6.3.1):

- o Type = 4: Admin Label. The Information field contains a free-form UTF-8 string whose byte length is given by the Information Length field. The value is administratively assigned. There is no requirement to terminate the string with null or any other character.

  Multiple admin labels can be included in the Peer Up notification. When multiple admin labels are included the BMP receiver MUST preserve their order.

  The TLV is optional.

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>.

[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>.

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/info/rfc7854>.

[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. URIs

[1] https://www.iana.org/assignments/bmp-parameters/bmp-parameters.xhtml

Acknowledgements

The authors would like to thank John Scudder and Mukul Srivastava for their valuable input.

Contributors

Manish Bhardwaj
Cisco Systems
3700 Cisco Way
San Jose, CA 95134
USA

Email: manbhard@cisco.com

Xianyuzheng
Tencent
Tencent Building, Kejizhongyi Avenue,
Hi-techPark, Nanshan District,Shenzhen 518057, P.R.China

Weiguo
Tencent
Tencent Building, Kejizhongyi Avenue,
Hi-techPark, Nanshan District,Shenzhen 518057, P.R.China

Shugang cheng
H3C

Authors' Addresses

Tim Evens
Cisco Systems
2901 Third Avenue, Suite 600
Seattle, WA 98121
USA

Email: tievens@cisco.com

Serpil Bayraktar
Cisco Systems
3700 Cisco Way
San Jose, CA 95134
USA

Email: serpil@cisco.com

Paolo Lucente
NTT Communications
Siriusdreef 70-72
Hoofddorp, WT 2132
NL

Email: paolo@ntt.net

Penghui Mi
Tencent
Tengyun Building,Tower A ,No. 397 Tianlin Road
Shanghai 200233
China

Email: kevinmi@tencent.com

Shunwan Zhuang
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: zhuangshunwan@huawei.com

Global Routing Operations
Internet-Draft
Updates: 7854 (if approved)
Intended status: Standards Track
Expires: 4 March 2022

T. Evens
S. Bayraktar
M. Bhardwaj
Cisco Systems
P. Lucente
NTT Communications

31 August 2021

Support for Local RIB in BGP Monitoring Protocol (BMP)
draft-ietf-grow-bmp-local-rib-13

Abstract

The BGP Monitoring Protocol (BMP) defines access to local Routing Information Bases (RIBs). This document updates BMP (RFC 7854) by adding access to the Local Routing Information Base (Loc-RIB), as defined in RFC 4271. The Loc-RIB contains the routes that have been selected by the local BGP speaker's Decision Process.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document defines a mechanism to monitor the BGP Loc-RIB state of remote BGP instances without the need to establish BGP peering sessions. BMP [RFC7854] does not define a method to send the BGP instance Loc-RIB. It does define in section 8.2 of [RFC7854] locally originated routes, but these routes are defined as the routes originated into BGP. For example, as defined by Section 9.4 of [RFC4271]. Loc-RIB includes all selected received routes from BGP peers in addition to locally originated routes.

Figure 1 shows the flow of received routes from one or more BGP peers into the Loc-RIB.

```
           +------------------+      +------------------+
           | Peer-A           |      | Peer-B           |
        /-- |                  | ---- |                  | --\
        |  | Adj-RIB-In (Pre) |      | Adj-RIB-In (Pre) |   |
        |  +------------------+      +------------------+   |
        |                 |                        |        |
        | Filters/Policy -|         Filters/Policy -|        |
        |                 V                        V        |
        |  +------------------+      +------------------+   |
        |  | Adj-RIB-In (Post)|      | Adj-RIB-In (Post)|   |
        |  +------------------+      +------------------+   |
        |                 |                        |        |
        |     Selected -|                Selected -|        |
        |                V                         V        |
        |   +----------------------------------------+      |
        |   |                 Loc-RIB                |      |
        |   +----------------------------------------+      |
        |                                                   |
        | ROUTER/BGP Instance                               |
        \---------------------------------------------------/
```

Figure 1: BGP peering Adj-RIBs-In into Loc-RIB

The following are some use-cases for Loc-RIB access:

* The Adj-RIB-In for a given peer Post-Policy may contain hundreds of thousands of routes, with only a handful of routes selected and installed in the Loc-RIB after best-path selection. Some monitoring applications, such as ones that need only to correlate flow records to Loc-RIB entries, only need to collect and monitor the routes that are actually selected and used.

Requiring the applications to collect all Adj-RIB-In Post-Policy data forces the applications to receive a potentially large unwanted data set and to perform the BGP decision process selection, which includes having access to the interior gateway protocol (IGP) next-hop metrics. While it is possible to obtain the IGP topology information using BGP Link-State (BGP-LS), it requires the application to implement shortest path first (SPF) and possibly constrained shortest path first (CSPF) based on additional policies. This is overly complex for such a simple application that only needs to have access to the Loc-RIB.

* It is common to see frequent changes over many BGP peers, but those changes do not always result in the router's Loc-RIB changing. The change in the Loc-RIB can have a direct impact on the forwarding state. It can greatly reduce time to troubleshoot and resolve issues if operators have the history of Loc-RIB changes. For example, a performance issue might have been seen for only a duration of 5 minutes. Post-facto troubleshooting this issue without Loc-RIB history hides any decision based routing changes that might have happened during those five minutes.

* Operators may wish to validate the impact of policies applied to Adj-RIB-In by analyzing the final decision made by the router when installing into the Loc-RIB. For example, in order to validate if multi-path prefixes are installed as expected for all advertising peers, the Adj-RIB-In Post-Policy and Loc-RIB needs to be compared. This is only possible if the Loc-RIB is available. Monitoring the Adj-RIB-In for this router from another router to derive the Loc-RIB is likely to not show same installed prefixes. For example, the received Adj-RIB-In will be different if ADD-PATH [RFC7911] is not enabled or if maximum supported number of equal paths is different between Loc-RIB and advertised routes.

This document adds Loc-RIB to the BGP Monitoring Protocol and replaces Section 8.2 of [RFC7854] Locally Originated Routes.

1.1. Alternative Method to Monitor Loc-RIB

Loc-RIB is used to build Adj-RIB-Out when advertising routes to a peer. It is therefore possible to derive the Loc-RIB of a router by monitoring the Adj-RIB-In Pre-Policy from another router. This becomes overly complex and error prone when considering the number of peers being monitored per router.

```
        /-----------------------------------------------------\
        |  ROUTER1 BGP Instance                               |
        |                                                     |
        |     +-------------------------------------------+   |
        |     |                 Loc-RIB                   |   |
        |     +-------------------------------------------+   |
        |                    |                   |            |
        |    +------------------+     +------------------+    |
        |    |   Peer-ROUTER2   |     |   Peer-ROUTER3   |    |
        |    | Adj-RIB-Out (Pre)|     | Adj-RIB-Out (Pre)|    |
        |    +------------------+     +------------------+    |
        |    Filters/Policy -|    Filters/Policy -|            |
        |                    V                    V            |
        |   +--------------------+     +--------------------+  |
        |   | Adj-RIB-Out (Post)|     | Adj-RIB-Out (Post)|    |
        |   +--------------------+     +--------------------+  |
        |              |                          |            |
        \------------- | ------------------------ | -----------/
                  BGP  |                     BGP  |
                  Peer |                     Peer |
           +------------------+          +------------------+
           |   Peer-ROUTER1   |          |   Peer-ROUTER1   |
        /--|                  |--\    /--|                  | --\
        |  | Adj-RIB-In (Pre) |  |    |  | Adj-RIB-In (Pre) |   |
        |  +------------------+  |    |  +------------------+   |
        |                        |    |                         |
        | ROUTER2/BGP Instance   |    | ROUTER3/BGP Instance    |
        \------------------------/    \-------------------------/
                    |                             |
                    v                             v
            ROUTER2 BMP Feed              ROUTER3 BMP Feed
```

Figure 2: Alternative method to monitor Loc-RIB

The setup needed to monitor the Loc-RIB of a router requires another router with a peering session to the target router that is to be monitored. As shown in Figure 2, the target router Loc-RIB is advertised via Adj-RIB-Out to the BMP router over a standard BGP peering session. The BMP router then forwards Adj-RIB-In Pre-Policy to the BMP receiver.

BMP lacking access to Loc-RIB introduces the need for additional resources:

* Requires at least two routers when only one router was to be monitored.

* Requires additional BGP peering to collect the received updates when peering may have not even been required in the first place. For example, virtual routing and forwarding (VRF) tables with no peers, redistributed BGP-LS with no peers, and segment routing egress peer engineering where no peers have link-state address family enabled are all situations with no preexisting BGP peers.

Many complexities are introduced when using a received Adj-RIB-In to infer a router Loc-RIB:

* Adj-RIB-Out received as Adj-RIB-In from another router may have a policy applied that filters, generates aggregates, suppresses more specific prefixes, manipulates attributes, or filters routes. Not only does this invalidate the Loc-RIB view, it adds complexity when multiple BMP routers may have peering sessions to the same router. The BMP receiver user is left with the error-prone task of identifying which peering session is the best representative of the Loc-RIB.

* BGP peering is designed to work between administrative domains and therefore does not need to include internal system level information of each peering router (e.g., the system name or version information). In order to derive the Loc-RIB of a router, the router name or other system information is needed. The BMP receiver and user are forced to do some type of correlation using what information is available in the peering session (e.g., peering addresses, autonomous system numbers, and BGP identifiers). This leads to error-prone correlations.

* Correlating BGP identifiers (BGP-ID) and session addresses to a router requires additional data, such as router inventory. This additional data provides the BMP receiver the ability to map and correlate the BGP-IDs and/or session addresses, but requires the BMP receiver to somehow obtain this data outside of BMP. How this data is obtained and the accuracy of the data directly affects the integrity of the correlation.

2. Terminology

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 RFC 2119 [RFC2119] RFC 8174 [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Definitions

* BGP Instance: refers to an instance of BGP-4 [RFC4271] and considerations in section 8.1 of [RFC7854] do apply to it.

* Adj-RIB-In: As defined in [RFC4271], "The Adj-RIBs-In contains unprocessed routing information that has been advertised to the local BGP speaker by its peers." This is also referred to as the pre-policy Adj-RIB-In in this document.

* Adj-RIB-Out: As defined in [RFC4271], "The Adj-RIBs-Out contains the routes for advertisement to specific peers by means of the local speaker's UPDATE messages."

* Loc-RIB: As defined in section 9.4 of [RFC4271], "The Loc-RIB contains the routes that have been selected by the local BGP speaker's Decision Process." Note that the Loc-RIB state as monitored through BMP might also contain routes imported from other routing protocols such as an IGP, or local static routes.

* Pre-Policy Adj-RIB-Out: The result before applying the outbound policy to an Adj-RIB-Out. This normally represents a similar view of the Loc-RIB but may contain additional routes based on BGP peering configuration.

* Post-Policy Adj-RIB-Out: The result of applying outbound policy to an Adj-RIB-Out. This MUST be what is actually sent to the peer.

4. Per-Peer Header

4.1. Peer Type

A new peer type is defined for Loc-RIB to distinguish that it represents the router Loc-RIB, which may have a route distinguisher (RD). Section 4.2 of [RFC7854] defines a Local Instance Peer type, which is for the case of non-RD peers that have an instance identifier.

This document defines the following new peer type:

* Peer Type = 3: Loc-RIB Instance Peer

4.2. Peer Flags

If locally sourced routes are communicated using BMP, they MUST be conveyed using the Loc-RIB instance peer type.

The per-peer header flags for Loc-RIB Instance Peer type are defined as follows:

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|F| | | | | | | |
+-+-+-+-+-+-+-+-+
```

* The F flag indicates that the Loc-RIB is filtered. This MUST be set when a filter is applied to Loc-RIB routes sent to the BMP collector.

  The unused bits are reserved for future use. They MUST be transmitted as 0 and their values MUST be ignored on receipt.

5. Loc-RIB Monitoring

The Loc-RIB contains all routes selected by the BGP Decision Process as described in section 9.1 of [RFC4271]. These routes include those learned from BGP peers via its Adj-RIBs-In Post-Policy, as well as routes learned by other means as per section 9.4 of [RFC4271]. Examples of these include redistribution of routes from other protocols into BGP or otherwise locally originated (i.e., aggregate routes).

As described in Section 6.1.2, a subset of Loc-RIB routes MAY be sent to a BMP collector by setting the F flag.

5.1. Per-Peer Header

All peer messages that include a per-peer header as defined in section 4.2 of [RFC7854] MUST use the following values:

* Peer Type: Set to 3 to indicate Loc-RIB Instance Peer.

* Peer Distinguisher: Zero filled if the Loc-RIB represents the global instance. Otherwise set to the route distinguisher or unique locally defined value of the particular instance the Loc-RIB belongs to.

* Peer Address: Zero-filled. Remote peer address is not applicable. The V flag is not applicable with Loc-RIB Instance peer type considering addresses are zero-filed.

* Peer AS: Set to the primary router BGP autonomous system number (ASN).

* Peer BGP ID: Set to the BGP instance global or RD (e.g., VRF) specific router-id section 1.1 of [RFC7854].

* Timestamp: The time when the encapsulated routes were installed in the Loc-RIB, expressed in seconds and microseconds since midnight (zero hour), January 1, 1970 (UTC). If zero, the time is unavailable. Precision of the timestamp is implementation-dependent.

5.2. Peer Up Notification

Peer Up notifications follow section 4.10 of [RFC7854] with the following clarifications:

* Local Address: Zero-filled, local address is not applicable.

* Local Port: Set to 0, local port is not applicable.

* Remote Port: Set to 0, remote port is not applicable.

* Sent OPEN Message: This is a fabricated BGP OPEN message. Capabilities MUST include the 4-octet ASN and all necessary capabilities to represent the Loc-RIB route monitoring messages. Only include capabilities if they will be used for Loc-RIB monitoring messages. For example, if ADD-PATH is enabled for IPv6 and Loc-RIB contains additional paths, the ADD-PATH capability should be included for IPv6. In the case of ADD-PATH, the capability intent of advertise, receive or both can be ignored since the presence of the capability indicates enough that add-paths will be used for IPv6.

* Received OPEN Message: Repeat of the same Sent Open Message. The duplication allows the BMP receiver to parse the expected received OPEN message as defined in section 4.10 of [RFC7854].

5.2.1. Peer Up Information

The following Peer Up information TLV type is added:

* Type = 3: VRF/Table Name. The Information field contains a UTF-8 string whose value MUST be equal to the value of the VRF or table name (e.g., RD instance name) being conveyed. The string size MUST be within the range of 1 to 255 bytes.

  The VRF/Table Name TLV is optionally included to support implementations that may not have defined a name. If a name is configured, it MUST be included. The default value of "global" MUST be used for the default Loc-RIB instance with a zero-filled distinguisher. If the TLV is included, then it MUST also be included in the Peer Down notification.

Multiple TLVs of the same type can be repeated as part of the same message, for example to convey a filtered view of a VRF. A BMP receiver should append multiple TLVs of the same type to a set in order to support alternate or additional names for the same peer. If multiple strings are included, their ordering MUST be preserved when they are reported.

5.3. Peer Down Notification

Peer Down notification MUST use reason code 6. Following the reason is data in TLV format. The following Peer Down information TLV type is defined:

* Type = 3: VRF/Table Name. The Information field contains a UTF-8 string whose value MUST be equal to the value of the VRF or table name (e.g., RD instance name) being conveyed. The string size MUST be within the range of 1 to 255 bytes. The VRF/Table Name informational TLV MUST be included if it was in the Peer Up.

5.4. Route Monitoring

Route Monitoring messages are used for initial synchronization of the Loc-RIB. They are also used to convey incremental Loc-RIB changes.

As defined in section 4.6 of [RFC7854], "Following the common BMP header and per-peer header is a BGP Update PDU."

5.4.1. ASN Encoding

Loc-RIB route monitor messages MUST use 4-byte ASN encoding as indicated in Peer Up sent OPEN message (Section 5.2) capability.

5.4.2. Granularity

State compression and throttling SHOULD be used by a BMP sender to reduce the amount of route monitoring messages that are transmitted to BMP receivers. With state compression, only the final resultant updates are sent.

For example, prefix 192.0.2.0/24 is updated in the Loc-RIB 5 times within 1 second. State compression of BMP route monitor messages results in only the final change being transmitted. The other 4 changes are suppressed because they fall within the compression interval. If no compression was being used, all 5 updates would have been transmitted.

A BMP receiver should expect that Loc-RIB route monitoring granularity can be different by BMP sender implementation.

5.5. Route Mirroring

Section 4.7 of [RFC7854], defines Route Mirroring for verbatim duplication of messages received. This is not applicable to Loc-RIB as PDUs are originated by the router. Any received Route Mirroring messages SHOULD be ignored.

5.6. Statistics Report

Not all Stat Types are relevant to Loc-RIB. The Stat Types that are relevant are listed below:

* Stat Type = 8: (64-bit Gauge) Number of routes in Loc-RIB.

* Stat Type = 10: Number of routes in per-AFI/SAFI Loc-RIB. The value is structured as: 2-byte AFI, 1-byte SAFI, followed by a 64-bit Gauge.

6. Other Considerations

6.1. Loc-RIB Implementation

There are several methods for a BGP speaker to implement Loc-RIB efficiently. In all methods, the implementation emulates a peer with Peer Up and Down messages to convey capabilities as well as Route Monitor messages to convey Loc-RIB. In this sense, the peer that conveys the Loc-RIB is a locally emulated peer.

6.1.1. Multiple Loc-RIB Peers

There MUST be at least one emulated peer for each Loc-RIB instance, such as with VRFs. The BMP receiver identifies the Loc-RIB by the peer header distinguisher and BGP ID. The BMP receiver uses the VRF/Table Name from the Peer Up information to associate a name to the Loc-RIB.

In some implementations, it might be required to have more than one emulated peer for Loc-RIB to convey different address families for the same Loc-RIB. In this case, the peer distinguisher and BGP ID should be the same since they represent the same Loc-RIB instance. Each emulated peer instance MUST send a Peer Up with the OPEN message indicating the address family capabilities. A BMP receiver MUST process these capabilities to know which peer belongs to which address family.

6.1.2. Filtering Loc-RIB to BMP Receivers

There maybe be use-cases where BMP receivers should only receive specific routes from Loc-RIB. For example, IPv4 unicast routes may include internal BGP (IBGP), external BGP (EBGP), and IGP but only routes from EBGP should be sent to the BMP receiver. Alternatively, it may be that only IBGP and EBGP that should be sent and IGP redistributed routes should be excluded. In these cases where the Loc-RIB is filtered, the F flag is set to 1 to indicate to the BMP receiver that the Loc-RIB is filtered. If multiple filters are associated to the same Loc-RIB, a Table Name MUST be used in order to allow a BMP receiver to make the right associations.

6.1.3. Changes to existing BMP sessions

In case of any change that results in the alteration of behavior of an existing BMP session, ie. changes to filtering and table names, the session MUST be bounced with a Peer Down/Peer Up sequence.

7. Security Considerations

The same considerations as in section 11 of [RFC7854] apply to this document. Implementations of this protocol SHOULD require that sessions are only established with authorized and trusted monitoring devices. It is also believed that this document does not add any additional security considerations.

8. IANA Considerations

This document requests that IANA assign the following new parameters to the BMP parameters name space (https://www.iana.org/assignments/bmp-parameters/bmp-parameters.xhtml).

8.1. BMP Peer Type

This document defines a new peer type (Section 4.1):

* Peer Type = 3: Loc-RIB Instance Peer

8.2. BMP Loc-RIB Instance Peer Flags

This document requests IANA to rename "BMP Peer Flags" to "BMP Peer Flags for Peer Types 0 through 2" and create a new registry named "BMP Peer Flags for Loc-RIB Instance Peer Type 3." This document defines that peer flags are specific to the Loc-RIB instance peer type. As defined in (Section 4.2):

* Flag 0: The F flag indicates that the Loc-RIB is filtered. This indicates that the Loc-RIB does not represent the complete routing table.

Flags 0 through 3 and 5 through 7 are unassigned. The registration procedure for the registry is "Standards Action".

8.3. Peer Up Information TLV

This document requests that IANA rename "BMP Initiation Message TLVs" registry to "BMP Initiation and Peer Up Information TLVs." section 4.4 of [RFC7854] defines that both Initiation and Peer Up share the same information TLVs. This document defines the following new BMP Peer Up information TLV type (Section 5.2.1):

* Type = 3: VRF/Table Name. The Information field contains a UTF-8 string whose value MUST be equal to the value of the VRF or table name (e.g., RD instance name) being conveyed. The string size MUST be within the range of 1 to 255 bytes.

8.4. Peer Down Reason code

This document defines the following new BMP Peer Down reason code (Section 5.3):

* Type = 6: Local system closed, TLV data follows.

8.5. Deprecated entries

This document also requests that IANA marks as "deprecated" the F Flag entry in the "BMP Peer Flags for Peer Types 0 through 2" registry.

9. 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>.

[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>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, DOI 10.17487/RFC5226, May 2008, <https://www.rfc-editor.org/info/rfc5226>.

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/info/rfc7854>.

[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. Informative References

[RFC7911] Walton, D., Retana, A., Chen, E., and J. Scudder, "Advertisement of Multiple Paths in BGP", RFC 7911, DOI 10.17487/RFC7911, July 2016, <https://www.rfc-editor.org/info/rfc7911>.

Acknowledgements

The authors would like to thank John Scudder, Jeff Haas and Mukul Srivastava for their valuable input.

Authors' Addresses

Tim Evens
Cisco Systems
2901 Third Avenue, Suite 600
Seattle, WA 98121
United States of America

Email: tievens@cisco.com

Serpil Bayraktar
Cisco Systems
3700 Cisco Way
San Jose, CA 95134
United States of America

Email: serpil@cisco.com

Manish Bhardwaj
Cisco Systems
3700 Cisco Way
San Jose, CA 95134
United States of America

Email: manbhard@cisco.com

Paolo Lucente
NTT Communications
Siriusdreef 70-72
2132 Hoofddorp
Netherlands

Email: paolo@ntt.net

Network Working Group
Internet-Draft
Updates: 7854 (if approved)
Intended status: Standards Track
Expires: 25 March 2024

J. Scudder
Juniper Networks

22 September 2023

Revision to Registration Procedures for Multiple BMP Registries
draft-ietf-grow-bmp-registries-change-04

Abstract

This document updates RFC 7854, BGP Monitoring Protocol (BMP) by making a change to the registration procedures for several registries. Specifically, any BMP registry with a range of 32768-65530 designated "Specification Required" has that range re-designated as "First Come First Served".

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

[RFC7854] creates a number of IANA registries that include a range of 32768-65530 designated "Specification Required". Each such registry also has a large range designated "Standards Action". Subsequent experience has shown two things. First, there is less difference between these two policies in practice than there is in theory (consider that [RFC8126] explains that for Specification Required, "Publication of an RFC is an ideal means of achieving this requirement"). Second, it's desirable to have a very low bar to registration, to avoid the risk of conflicts introduced by use of unregistered code points (so-called "code point squatting").

Accordingly, this document revises the registration procedures, as given in Section 2.

2. IANA Considerations

IANA is requested to revise the following registries within the BMP group:

* BMP Statistics Types
* BMP Initiation Message TLVs
* BMP Termination Message TLVs
* BMP Termination Message Reason Codes
* BMP Route Mirroring TLVs
* BMP Route Mirroring Information Codes

For each of these registries, the ranges 32768-65530 whose registration procedures were "Specification Required" are revised to have the registration procedures "First Come First Served".

3. Security Considerations

This revision to registration procedures does not change the underlying security issues inherent in the existing [RFC7854].

4. Acknowledgements

Thanks to Jeff Haas for review and encouragement, and to Tom Petch for review.

5. Normative References

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/info/rfc7854>.

[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, <https://www.rfc-editor.org/info/rfc8126>.

Author's Address

John Scudder
Juniper Networks
1194 N. Mathilda Ave
Sunnyvale, CA 94089
United States of America
Email: jgs@juniper.net

Global Routing Operations
Internet-Draft
Intended status: Informational
Expires: October 26, 2020

J. Snijders
NTT
M. Stucchi
Independent
M. Aelmans
Juniper Networks

April 24, 2020

RPKI Autonomous Systems Cones: A Profile To Define Sets of Autonomous Systems Numbers To Facilitate BGP Filtering

draft-ietf-grow-rpki-as-cones-02

Abstract

This document describes a way to define groups of Autonomous System numbers in RPKI [RFC6480]. We call them AS-Cones. AS-Cones provide a mechanism to be used by operators for filtering BGP-4 [RFC4271] announcements.

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

Table of Contents

1. Introduction

The main goal of the Resource Public Key Infrastructure (RPKI) system [RFC6480] is to support improved security for the global routing system. This is achieved through the use of information stored in a distributed repository system comprised of signed objects. A

commonly used object type is the Route Object Authorisation (ROAs), which describe the relation between a prefix and its originating ASNs.

There is however no method for an operator to assert the routes for its customer networks, making it difficult to use the information carried by RPKI to create meaningful BGP-4 filters without relying on RPSL [RFC2622] as-sets.

This document introduces a new attestation object, called an AS-Cone. An AS-Cone is a digitally signed object with the goal to enable operators to define a set of customer or downstream ASNs that can be found as "right adjacencies", or transit customer networks, facilitating the construction of prefix filters for a given ASN, thus making routing more secure.

The goal of AS-Cones is to be able to recursively define all the originating ASNs that define the customer base of a given ASN, including all the transit relationships. This means that through AS-Cones, it is possible to create a tree of all the neighbour relationships for the customers of a given Autonomous System.

2. Format of AS-Cone objects

AS-Cones are composed of two types of distinct objects:

- o Policy definitions; and
- o The AS-Cones themselves.

These objects are stored in ASN.1 format and are digitally signed according to the same rules and conventions applied for RPKI ROA Objects ([RFC6482]).

2.1. Policy definition object

A policy definition object contains a list of the upstream and peering relationships for a given Autonomous System that need an AS-Cone to be used for filtering. For each relationship, either an AS-Cone or a plain Autonomous System Number is referenced to indicate which networks will be announced to the other end of the relationship using BGP.

The default behaviour for a neighbour, if the relationship is not explicitly described in the policy, is to only accept the networks originated by the ASN. This means that a stub ASN neither has to set up any AS-Cone, description, nor policy.

The Policy Definition object contains a field called "ContactEmail" containing the E-Mail address for which all the communication related to this policy definition should be sent to.

Only one AS-Cone or Autonomous System Number can be supplied for a given relationship. If more than one AS-Cone needs to be announced in the relationship, then it is mandatory to create a third AS-Cone that includes those two. If more than one ASN needs to be referenced, then an AS-Cone for the relationship needs to be created.

2.1.1. Naming convention for Policy definition objects

A Policy object is referenced using the Autonomous System number it refers to, preceded by the string "AS".

2.1.2. ASN.1 format of a Policy Definition object

```
ASNPolicy DEFINITIONS ::=
BEGIN
Neighbours ::= SEQUENCE OF Neighbour

Neighbour ::= SEQUENCE
{
ASN INTEGER (1..42949672965),
ASCone  VisibleString
}

Version ::= INTEGER
LastModified ::= GeneralizedTime
Created ::= GeneralizedTime
ContactEmail ::= PrintableString(SIZE (1..75))
END
```

ASN.1 format of a Policy definition object

2.1.3. Naming convention for neighbour relationships

When referring to a neighbour relationship contained in a Policy definition object, the following convention should be used:

ASX:ASY

Where X is the number of the ASN holder and Y is the number of the ASN intended to use the AS-Cone object to generate a filter.

2.2. AS-Cone definition object

An AS-Cone contains a list of the downstream customer ASNs and AS-Cones of a given ASN. The list is used to create filter lists by the networks providing transit to or having a peering relationship with the ASN.

An AS-Cone can reference another AS-Cone.

2.2.1. Adding entries in an AS-Cone object

When an entry is added, it is in the Unverified status, and its "Verified" variable is set to 0.

If an ASN is added as an entry, it becomes directly visible and usable in building prefix lists, and a notification is sent to the E-mail address contained in the "ContactEmail" field of the AS-Cone Policy Object for that Autonomous System Number. The holder of the Autonomous System Number can acknowledge the notification, in which case the "Verified" field is switched to the value of 1.

If an AS-Cone is added to the object, a notification is sent to the E-Mail address contained in the "ContactEmail" field of the AS-Cone object that is being added. If the "ContactEmail" field is blank, the notification is sent to the E-mail address contained in the "ContactEmail" field of the AS-Cone Policy Object of the ASN of which the AS-Cone is part of. Only when an acknowledgement from the holder of the object is obtained, the "Verified" field is changed to a value of 1, and the AS-Cone becomes visible.

The value of the "Verified" field is fundamental for the creation of appropriate prefix filtering rules as described later.

2.2.2. Removal of entries from an AS-Cone object

The owner of an AS-Cone can remove any entry from its object without requesting any permission from the holders of the entries being removed.

The holder of an entry in a third party AS-Cone can remove the entry by performing authentication based on the E-mail address contained in the "ContactEmail" field of the resource itself. The RIRs MUST provide means to perform this authentication via an auth code, an API, or other means. The removal of an entry SHOULD be immediate upon successful authentication.

2.2.3. Naming convention for AS-Cone objects

AS-Cones MUST have a unique name for the ASN they belong to. Names are composed of ASCII strings up to 255 characters long and cannot contain spaces.

In order for AS-Cones to be unique in the global routing system, their string name is preceded by the AS number of the ASN they are part of, followed by ":". For example, AS-Cone "EuropeanCustomers" for ASN 65530 is represented as "AS65530:EuropeanCustomers" when referenced from a third party.

2.2.4. ASN.1 format of an AS-Cone

```
ASCone DEFINITIONS ::=
BEGIN
Entities ::= SEQUENCE OF Entity

Entity CHOICE
{
        ASN INTEGER (1..4294967295),
        OtherASCone VisibleString
        Verified ::= BOOLEAN
}

Version ::= INTEGER
LastModified ::= GeneralizedTime
Created ::= GeneralizedTime
ContactEmail ::= PrintableString(SIZE (1..75))
END
```

ASN.1 format of an AS-Cone

3. Validating an AS-Cone

In order to validate a full AS-Cone, a network operator MUST have access to the validated cache of an RPKI validator software containing all the Policy definition and AS-Cone objects. Validation occurs following the description in [RFC6488]:

In order to validate a full AS-Cone, an operator SHOULD perform the following steps:

1. For every downstream ASN, the operator verifies if a related policy definition (see Section 2.1) file exists. If no object exists, the status of the AS-Cone is "Unknown". If instead it

exists, it proceeds to collect a list of ASNs for the cone by looking at the following data, in exact order:

1. A policy for the specific relationship, in the form of ASX:ASY, where ASX is the downstream ASN, and ASY is the ASN of the operator validating the AS-Cone;

2. If there is no specific definition for the relationship, the ASX:Default policy;

If none of the two definitions above exists, then the operator should only consider the ASN of its downstream to be added to the list.

2. These objects can either point to:

   1. An AS-Cone; or

   2. An ASN

3. If the definition points to an AS-Cone, the operator looks for the object referenced, which should be contained in the validated cache;

4. If the validated cache does not contain the referenced object, then the validation moves on to the next downstream ASN;

5. If the validated cache contains the referenced object, the validation process evaluates every entry in the AS-Cone. For each entry:

   1. If there is a reference to an ASN, then the operator adds the ASN to the list for the given AS-Cone;

   2. If there is a reference to another AS-Cone, the validating process should recursively process all the entries in that AS-Cone first, with the same principles contained in this list.

   Since the goal is to build a list of ASNs announcing routes in the AS-Cone, then if an ASN or an AS-Cone are referenced more than once in the process, their contents should only be added once to the list. This is intended to avoid endless loops, and in order to avoid cross-reference of AS-Cones.

6. When all the AS-Cones referenced in the policies have been recursively iterated, and all the originating ASNs have been taken into account, the operator can then build a full prefix-

list with all the prefixes originated in its AS-Cone. This can be done by querying the RPKI validator software for all the networks originated by every ASN referenced in the AS-Cone.

4. Types of validation for AS-Cones

AS-Cones can be validated in 4 different ways:

Loose Validation. This is the method described in the procedure above;

Opportunistic Validation. This is similar to Loose validation, but it discards all the ASNs for which the "Validated" fields have a value of 0. The intent is to remove from the prefix list all the ASNs that haven't validated their entry in the customer cone for the operator;

Almost-Strict validation. In this method, whenever an entry with the "Validated" field set to 0 is found, the entire sub-tree (the AS-Cone) in which it is contained is discarded.

Strict Validation. In this method, only the entries with the "Validated" field set to 1 are considered. If even a single entry has a "Validated" field set to 0, the whole AS-Cone is discarded.

It is important to note that no AS-Cone with the "Validated" field set to 0 is going to be visible at any time, so they are automatically discarded. This protects AS-Cone holders from being considered customers of a third party without their consent.

5. Recommendations for use of AS-Cones at Internet Exchange points

When an operator is a member of an internet exchange point, it is recommended for it to create at least a Default policy.

In case of a peering session with a route server, the operator could publish a policy pointing to the ASN of the route server. A route server operator, then, could build strict prefix filtering rules for all the participants, and offer it as a service to its members.

For internet exchange points operators, the recommendation is to use Strict Filtering as explained in the previous section.

6. Publication of AS-Cones as IRR objects

AS-Cones are very similar to AS-Set RPSL Objects, so they could also be published in IRR Databases as AS-Set objects. Every ASN contained in an AS-Cone, and all the AS-Cones referenced should be considered

as member: attributes. The naming convention for AS-Cones (ASX:AS-Cone) should be maintained, in order to keep consistency between the two databases.

7. Security Considerations

TBW

8. IANA Considerations

This memo includes no request to IANA.

9. Contributors

The following people contributed significantly to the content of the document: Greg Skinner.

10. Acknowledgments

The authors would like to thank Randy Bush, Nick Hilliard and Aftab Siddiqui.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[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>.

[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>.

11.2. Informative References

[RFC2622] Alaettinoglu, C., Villamizar, C., Gerich, E., Kessens, D., Meyer, D., Bates, T., Karrenberg, D., and M. Terpstra, "Routing Policy Specification Language (RPSL)", RFC 2622, DOI 10.17487/RFC2622, June 1999, <https://www.rfc-editor.org/info/rfc2622>.

[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>.

[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>.

Authors' Addresses

Job Snijders
NTT Ltd.
Theodorus Majofskistraat 100
Amsterdam 1065 SZ
The Netherlands

Email: job@ntt.net

Massimiliano Stucchi
Independent

Email: max@stucchi.ch

Melchior Aelmans
Juniper Networks
Boeing Avenue 240
Schiphol-Rijk 1119 PZ
The Netherlands

Email: maelmans@juniper.net

Network Working Group
Internet-Draft
Updates: 1997 (if approved)
Intended status: Standards Track
Expires: December 15, 2019

J. Borkenhagen
AT&T
R. Bush
IIJ & Arrcus
R. Bonica
Juniper Networks
S. Bayraktar
Cisco Systems
June 13, 2019

Well-Known Community Policy Behavior
draft-ietf-grow-wkc-behavior-08

Abstract

Well-Known BGP Communities are manipulated differently across various current implementations; resulting in difficulties for operators. Network operators should deploy consistent community handling across their networks while taking the inconsistent behaviors from the various BGP implementations into consideration.. This document recommends specific actions to limit future inconsistency, namely BGP implementors must not create further inconsistencies from this point forward.

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 will expire on December 15, 2019.

Copyright Notice

Table of Contents

1. Introduction

The BGP Communities Attribute was specified in [RFC1997] which introduced the concept of Well-Known Communities. In hindsight, [RFC1997] did not prescribe as fully as it should have how Well-Known Communities may be manipulated by policies applied by operators. Currently, implementations differ in this regard, and these differences can result in inconsistent behaviors that operators find difficult to identify and resolve.

This document describes the current behavioral differences in order to assist operators in generating consistent community-manipulation policies in a multi-vendor environment, and to prevent the introduction of additional divergence in implementations.

This document recommends specific actions to limit future inconsistency, namely BGP implementors MUST NOT create further inconsistencies from this point forward.

2. Manipulation of Communities by Policy

[RFC1997] says:

"A BGP speaker receiving a route with the COMMUNITIES path attribute may modify this attribute according to the local policy."

One basic operational need is to add or remove one or more communities to the set. The focus of this document is another common operational need, to replace all communities with a new set. To simplify this second case, most BGP policy implementations provide syntax to "set" community that operators use to mean "remove any/all communities present on the route, and apply this set of communities instead."

Some operators prefer to write explicit policy to delete unwanted communities rather than using "set;" i.e. using a "delete community *:*" and then "add community x:y ..." configuration statements in an attempt to replace all communities. The same community manipulation policy differences described in the following section exist in both "set" and "delete community *:*" syntax. For simplicity, the remainder of this document refers only to the "set" behaviors, which we refer to collectively as each implementation's '"set" directive.'

3. Community Manipulation Policy Differences

Vendor implementations differ in the treatment of certain Well-Known communities when modified using the syntax to "set" the community. Some replace all communities including the Well-Known ones with the new set, while others replace all non-Well-Known Communities but do not modify any Well-Known Communities that are present.

These differences result in what would appear to be identical policy configurations having very different results on different platforms.

4. Documentation of Vendor Implementations

In this section we document the syntax and observed behavior of the "set" directive in several popular BGP implementations to illustrate the severity of the problem operators face.

In Juniper Networks' Junos OS, "community set" removes all communities, Well-Known or otherwise.

In Cisco IOS XR, "set community" removes all communities except for the following:

Numeric	Common Name
0:0	internet
65535:0	graceful-shutdown
65535:1	accept-own rfc7611
65535:65281	NO_EXPORT
65535:65282	NO_ADVERTISE
65535:65283	NO_EXPORT_SUBCONFED (or local-AS)

Communities not removed by Cisco IOS XR

Table 1

Cisco IOS XR does allow Well-Known communities to be removed only by explicitly enumerating one at a time, not in the aggregate; for example, "delete community accept-own". Operators are advised to consult Cisco IOS XR documentation and/or Cisco support for full details.

On Extreme networks' Brocade NetIron: "set community X" removes all communities and sets X.

In Huawei's VRP product, "community set" removes all communities, Well-Known or otherwise.

In OpenBGPD, "set community" does not remove any communities, Well-Known or otherwise.

Nokia's SR OS has several directives that operate on communities. Its "set" directive is called using the "replace" keyword, replacing all communities, Well-Known or otherwise, with the specified communities.

4.1. Note on an Inconsistency

The IANA publishes a list of Well-Known Communities [IANA-WKC].

Cisco IOS XR's set of Well-Known communities that "set community" will not overwrite diverges from the IANA's list of Well-Known communities. Quite a few Well-Known communities from IANA's list do not receive special treatment in Cisco IOS XR, and at least one community on Cisco IOS XR's special treatment list, internet == 0:0,

is not formally a Well-Known Community as it is not in [IANA-WKC]; but taken from the Reserved range [0x00000000-0x0000FFFF].

This merely notes an inconsistency. It is not a plea to 'protect' the entire IANA list from "set community."

5. Note for Those Writing RFCs for New Community-Like Attributes

When establishing new [RFC1997]-like attributes (large communities, wide communities, etc.), RFC authors should state explicitly how the new attribute is to be handled.

6. Action Items

Network operators are encouraged to limit their use of the "set" directive (within reason), to improve consistency across platforms.

Unfortunately, it would be operationally disruptive for vendors to change their current implementations.

Vendors MUST clearly document the behavior of "set" directive in their implementations.

Vendors MUST ensure that their implementations' "set" directive treatment of any specific community does not change if/when that community becomes a new Well-Known Community through future standardization. For most implementations, this means that the "set" directive MUST continue to remove the community; for those implementations where the "set" directive removes no communities, that behavior MUST continue.

Given the implementation inconsistencies described in this document, network operators are urged never to rely on any implicit understanding of a neighbor ASN's BGP community handling. I.e., before announcing prefixes with NO_EXPORT or any other community to a neighbor ASN, the operator should confirm with that neighbor how the community will be treated.

7. Security Considerations

Surprising defaults and/or undocumented behaviors are not good for security. This document attempts to remedy that.

8. IANA Considerations

The IANA is requested to list this document as an additional reference for the [IANA-WKC] registry.

9. Acknowledgments

The authors thank Martijn Schmidt, Qin Wu for the Huawei data point, Greg Hankins, Job Snijders, David Farmer, John Heasley, and Jakob Heitz.

10. Normative References

[IANA-WKC]
IANA, "Border Gateway Protocol (BGP) Well-Known Communities", <https://www.iana.org/assignments/bgp-well-known-communities>.

[RFC1997] Chandra, R., Traina, P., and T. Li, "BGP Communities Attribute", RFC 1997, DOI 10.17487/RFC1997, August 1996, <http://www.rfc-editor.org/info/rfc1997>.

[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>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <http://www.rfc-editor.org/info/rfc8174>.

Authors' Addresses

Jay Borkenhagen
AT&T
200 Laurel Avenue South
Middletown, NJ 07748
United States of America

Email: jayb@att.com

Randy Bush
IIJ & Arrcus
5147 Crystal Springs
Bainbridge Island, WA 98110
US

Email: randy@psg.com

Ron Bonica
Juniper Networks
2251 Corporate Park Drive
Herndon, VA 20171
US

Email: rbonica@juniper.net

Serpil Bayraktar
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
United States of America

Email: serpil@cisco.com

Global Routing Operations
Internet-Draft
Intended status: Standards Track
Expires: September 9, 2019

J. Snijders
NTT Communications
M. Aelmans
Juniper Networks

March 8, 2019

BGP Maximum Prefix Limits
draft-sa-grow-maxprefix-02

Abstract

This document describes mechanisms to limit the negative impact of route leaks [RFC7908] and/or resource exhaustion in BGP [RFC4271] implementations.

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

Copyright Notice

Table of Contents

1. Introduction

This document describes mechanisms to reduce the negative impact of certain types of misconfigurations and/or resource exhaustions in BGP [RFC4271] operations. While [RFC4271] already described a method to tear down BGP sessions when certain thresholds are exceeded, some nuances in this specification were missing resulting in inconsistencies between BGP implementations. In addition to clarifying "inbound maximum prefix limits", this document also introduces a specification for "outbound maximum prefix limits".

2. Inbound Maximum Prefix Limits

An operator MAY configure a BGP speaker to terminate its BGP session with a neighbor when the number of address prefixes received from that neighbor exceeds a locally configured upper limit. The BGP speaker then MUST send the neighbor a NOTIFICATION message with the Error Code Cease and the Error Subcode "Threshold reached: Maximum Number of Prefixes Received", and MAY support other actions. Reporting when thresholds have been exceeded is an implementation specific consideration, but SHOULD include methods such as Syslog

[RFC5424]. Inbound Maximum Prefix Limits can be applied in two distinct places in the conceptual model: before or after the application of routing policy.

2.1. Type A: Pre-Policy Inbound Maximum Prefix Limits

The Adj-RIBs-In stores routing information learned from inbound UPDATE messages that were received from another BGP speaker Section 3.2 [RFC4271]. The Type A pre-policy limit uses the number of NLRIs per Address Family Identifier (AFI) per Subsequent Address Family Identifier (SAFI) as input into its threshold comparisons. For example, when an operator configures the Type A pre-policy limit for IPv4 Unicast to be 50 on a given EBGP session, and the other BGP speaker announces its 51st IPv4 Unicast NLRI, the session MUST be terminated.

Type A pre-policy limits are particularly useful to help dampen the effects of full table route leaks and memory exhaustion when the implementation stores rejected routes.

2.2. Type B: Post-Policy Inbound Maximum Prefix Limits

RFC4271 describes a Policy Information Base (PIB) that contains local policies that can be applied to the information in the Routing Information Base (RIB). The Type B post-policy limit uses the number of NLRIs per Address Family Identifier (AFI) per Subsequent Address Family Identifier (SAFI), after application of the Import Policy as input into its threshold comparisons. For example, when an operator configures the Type B post-policy limit for IPv4 Unicast to be 50 on a given EBGP session, and the other BGP speaker announces a hundred IPv4 Unicast routes of which none are accepted as a result of the local import policy (and thus not considered for the Loc-RIB by the local BGP speaker), the session is not terminated.

Type B post-policy limits are useful to help prevent FIB exhaustion and prevent accidental BGP session teardown due to prefixes not accepted by policy anyway.

3. Outbound Maximum Prefix Limits

An operator MAY configure a BGP speaker to terminate its BGP session with a neighbor when the number of address prefixes to be advertised to that neighbor exceeds a locally configured upper limit. The BGP speaker then MUST send the neighbor a NOTIFICATION message with the Error Code Cease and the Error Subcode "Threshold reached: Maximum Number of Prefixes Send", and MAY support other actions. Reporting when thresholds have been exceeded is an implementation specific

consideration, but SHOULD include methods such as Syslog [RFC5424]. By definition, Outbound Maximum Prefix Limits are Post-Policy.

The Adj-RIBs-Out stores information selected by the local BGP speaker for advertisement to its neighbors. The routing information stored in the Adj-RIBs-Out will be carried in the local BGP speaker's UPDATE messages and advertised to its neighbors Section 3.2 [RFC4271]. The Outbound Maximum Prefix Limit uses the number of NLRIs per Address Family Identifier (AFI) per Subsequent Address Family Identifier (SAFI), after application of the Export Policy, as input into its threshold comparisons. For example, when an operator configures the Outbound Maximum Prefix Limit for IPv4 Unicast to be 50 on a given EBGP session, and were about to announce its 51st IPv4 Unicast NLRI to the other BGP speaker as a result of the local export policy, the session MUST be terminated.

Outbound Maximum Prefix Limits are useful to help dampen the negative effects of a misconfiguration in local policy. In many cases, it would be more desirable to tear down a BGP session rather than causing or propagating a route leak.

4. Considerations for Operations with Multi-Protocol BGP

5. Considerations for soft thresholds

describe soft and hard limits (warning vs teardown)

6. Security Considerations

Maximum Prefix Limits are an essential tool for routing operations and SHOULD be used to increase stability.

7. IANA Considerations

This memo requests that IANA updates the name of subcode "Maximum Number of Prefixes Reached" to "Threshold exceeded: Maximum Number of Prefixes Received" in the "Cease NOTIFICATION message subcodes" registry under the "Border Gateway Protocol (BGP) Parameters" group.

This memo requests that IANA assigns a new subcode named "Threshold exceeded: Maximum Number of Prefixes Send" in the "Cease NOTIFICATION message subcodes" registry under the "Border Gateway Protocol (BGP) Parameters" group.

8. Acknowledgments

The authors would like to thank Saku Ytti and John Heasley (NTT Communications), Jeff Haas, Colby Barth and John Scudder (Juniper Networks), Martijn Schmidt (i3D.net), Teun Vink (BIT), Sabri Berisha (eBay), Martin Pels (Quanza), Steven Bakker (AMS-IX), Aftab Siddiqui (ISOC) and Yu Tianpeng for their support, insightful review, and comments.

9. Implementation status - RFC EDITOR: REMOVE BEFORE PUBLICATION

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.

The below table provides an overview (as of the moment of writing) of which vendors have produced implementation of inbound or outbound maximum prefix limits. Each table cell shows the applicable configuration keywords if the vendor implemented the feature.

Vendor	Type A Pre-Policy	Type B Post-Policy	Outbound
Cisco IOS XR		maximum-prefix	
Cisco IOS XE		maximum-prefix	
Juniper Junos OS	prefix-limit	accepted-prefix-limit, or prefix-limit combined with 'keep none'	
Nokia SR OS	prefix-limit		
NIC.CZ BIRD	'import keep filtered' combined with 'receive limit'	'import limit' or 'receive limit'	export limit
OpenBSD OpenBGPD	max-prefix		
Arista EOS	maximum-routes	maximum-accepted-routes	
Huawei VRPv5	peer route-limit		
Huawei VRPv8	peer route-limit	peer route-limit accept-prefix	

First presented by Snijders at [RIPE77]

Table 1: Maximum prefix limits capabilities per implementation

10. Appendix: Implementation Guidance

1) make it clear who does what: if A sends too many prefixes to B A should see "ABC" in log B should see "DEF" in log to make it clear which of the two parties does what 2) recommended by default automatically restart after between 15 and 30 minutes

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[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>.

[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>.

11.2. Informative References

[RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424, DOI 10.17487/RFC5424, March 2009, <https://www.rfc-editor.org/info/rfc5424>.

[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>.

[RIPE77] Snijders, J., "Robust Routing Policy Architecture", May 2018, <https://ripe77.ripe.net/wp-content/uploads/presentations/59-RIPE77_Snijders_Routing_Policy_Architecture.pdf>.

Authors' Addresses

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

Email: job@ntt.net

Melchior Aelmans
Juniper Networks
Boeing Avenue 240
Schiphol-Rijk 1119 PZ
The Netherlands

Email: maelmans@juniper.net

GROW
Internet-Draft
Updates: 7854 (if approved)
Intended status: Standards Track
Expires: June 17, 2019

J. Scudder
Juniper Networks
December 14, 2018

BMP Peer Up Message Namespace
draft-scudder-grow-bmp-peer-up-00.txt

Abstract

RFC 7854, BMP, uses different message types for different purposes. Most of these are Type, Length, Value (TLV) structured. One message type, the Peer Up message, lacks a set of TLVs defined for its use, instead sharing a namespace with the Initiation message. Subsequent experience has shown that this namespace sharing was a mistake, as it hampers the extension of the protocol.

This document updates RFC 7854 by creating an independent namespace for the Peer Up message. The changes in this document are formal only, compliant implementations of RFC 7854 also comply with this specification.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

[RFC7854] defines a number of different BMP message types. With the exception of the Route Monitoring message type, these messages are TLV-structured. Most message types have distinct namespaces and IANA registries. However, the namespace of the Peer Up message overlaps that of the Initiation message. As the BMP protocol has been extended, this oversight has become problematic. In this document, we create a distinct namespace for the Peer Up message to eliminate this overlap, and create the corresponding missing registry.

The changes in this document are formal only, compliant implementations of [RFC7854] also comply with this specification.

1.1. Requirements Language

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

2. String Definition

A string TLV is a free-form sequence of UTF-8 characters whose length is given by the TLV's Length field. There is no requirement to terminate the string with a null (or any other particular) character -- the Length field gives its termination.

3. Changes to RFC 7854

We update [RFC7854] as follows:

- o The "Information TLV" of section 4.4, that was shared between the Initiation and Peer Up message types, is renamed as the "Initiation Information TLV", and is only relevant to the Initiation message type.

- o A "Peer Up Information TLV" is defined, and is relevant to the Peer Up message type.

- o A "Peer Up TLVs" registry is created, seeded with the Peer Up Information TLV.

Other than as summarized above, and detailed below, there are no other changes.

3.1. Revision to Information TLV, Renamed as Initiation Information TLV

The Information TLV defined in section 4.4 of [RFC7854] is renamed "Initiation Information TLV". It is used only by the Initiation message, not by the Peer Up message.

The definition of Type = 0 is revised to be:

- o Type = 0: String. The Information field contains a string (Section 2). The value is administratively assigned. If multiple strings are included, their ordering MUST be preserved when they are reported.

3.2. Revision to Peer Up Notification

The final paragraph of section 4.10 of [RFC7854] references the Information TLV (which is revised above (Section 3.1)). That paragraph is replaced by the following:

- o Information: Information about the peer, using the Peer Up Information TLV format defined below (Section 3.3). The String type may be repeated. Inclusion of the Information field is

OPTIONAL. Its presence or absence can be inferred by inspection of the Message Length in the common header.

3.3. Definition of Peer Up Information TLV

The Peer Up Information TLV is used by the Peer Up message.

```
 0 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Information Type     |       Information Length      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Information (variable)                        |
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- o Information Type (2 bytes): Type of information provided. Defined types are:

  - * Type = 0: String. The Information field contains a string (Section 2). The value is administratively assigned. If multiple strings are included, their ordering MUST be preserved when they are reported.

- o Information Length (2 bytes): The length of the following Information field, in bytes.

- o Information (variable): Information about the monitored router, according to the type.

4. IANA Considerations

IANA is requested to create a registry within the BMP group, named "BMP Peer Up Message TLVs", reference this document.

Registration procedures for this registry are:

Range	Registration Procedures
0-32767	Standards Action
32768-65530	First Come, First Served
65531-65534	Experimental
65535	Reserved

Initial values for this registry are:

Type	Description	Reference
0	String	this document
65535	Reserved	this document

5. Security Considerations

This rearrangement of deck chairs does not change the underlying security issues inherent in the existing [RFC7854].

6. Acknowledgements

TBD

7. 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>.

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/info/rfc7854>.

[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>.

Author’s Address

John Scudder
Juniper Networks
1194 N. Mathilda Ave
Sunnyvale, CA 94089
USA

Email: jgs@juniper.net

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: August 10, 2019

R. Szarecki, Ed.
K. Vairavakkalai
N. Venkataraman
Juniper Networks Inc.

February 6, 2019

Use of Abstract NH in Scale-Out peering architecture
draft-szarecki-grow-abstract-nh-scaleout-peering-00

Abstract

Many large-scale service provider networks use some form of scale-out architecture at peering sites. In such an architecture, each participating Autonomous System (AS) deploys multiple independent Autonomous System Border Routers (ASBRs) for peering, and Equal Cost Multi-Path (ECMP) load balancing is used between them. There are numerous benefits to this architecture, including but not limited to N+1 redundancy and the ability to flexibly increase capacity as needed. A cost of this architecture is an increase in the amount of state in both the control and data planes. This has negative consequences for network convergence time and scale.

In this document we describe how to mitigate these negative consequences through configuration of the routing protocols, both BGP and IGP, to utilize what we term the "Abstract Next-Hop" (ANH). Use of ANH allows us to both reduce the number of BGP paths in the control plane and enable rapid path invalidation (hence, network convergence and traffic restoration). We require no new protocol features to achieve these benefits.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Common to all large Internet networks are the requirements for large aggregate bandwidth and low latency. As network sizes and traffic volumes have increased, it has become common to use scale-out architectures to satisfy these requirements. Use of these techniques within individual networks is well-known. Here, we explore a scale-out architecture for interconnecting different Autonomous Systems (ASes).

Below, we show an example topology. Content is hosted within AS 2, consumers connect via the various ISP Metro ASes.

```
+---------------+      +----------------+       +---------------+
|               |      |                +-------+               |
|               +------+                +-------+      AS 30    |
|               +------+                |       |   ISP Metro   |
|               +------+                |  /----+               |
|               |      |                | //----+               |
|      AS 2     |      |       AS 1     |//     +---------------+
|    Content    |      |  ISP BackBone  X/
|   provider    +------+                X\
|               +------+                |\\     +---------------+
|               |      |                | \\----+               |
|               |      |                |  \----+     AS 31     |
|               +------+                |       |   ISP Metro   |
|               +------+                +-------+               |
|               +------+                +-------+               |
+---------------+      +----------------+       +---------------+
```

Figure 1

ASes 1 and 2 are connected at multiple, geographically diverse, sites. Geographic diversity is required for reasons including resiliency, minimization of latency, and minimization of cost associated with long-distance data transmission.

1.1. Scale-Out peering

The same trends that have driven the use of scale-out architectures within ASes drive interest in using them at peering sites. In such an architecture, each AS at the peering site deploys multiple independent Autonomous System Border Routers (ASBRs). Benefits that can be realized include N+1 redundancy and the ability to flexibly increase capacity as needed. The ASBRs are often connected to the rest of their AS in a leaf-spine topology through core routers, and augmented with a per-site pair of BGP route reflectors (RRs). See for example SITE1 in Figure 2, below.

The fundamental requirements in this architecture are:

a. Keep traffic on a path that has low latency.

b. Utilize all peering links that offer low latency.

c. In the event of failure, minimize the time needed to restore service.

1.1.1. Low latency

BGP, the Border Gateway Protocol, does not directly carry delay information. We make the general assumption in this document that paths selected by the BGP best path algorithm [RFC4271] will provide lower latency than those not selected. This assumption is not guaranteed to be true, but lacking special arrangements between peering ASes, it is what the protocol is able to provide.

1.1.2. All equal cost paths utilization

In order to use all links between peering ASes that provide the same BGP path costs to the destination prefix, at a minimum BGP speakers need to be enabled for multi-path operation. Additionally, all AS ingress BGP speakers need to know at least all equal and best paths to the destination via multiple ASBRs. If a full IBGP mesh is used, this happens naturally. However, IBGP full meshes are uncommon in large networks and are even more impractical in scale-out architectures due to the high total number of ASBRs.

The well-known techniques to deal with full-mesh scale challenges - Route Reflection [RFC4456] and Confederations [RFC5065] - hide redundant paths, as they advertise only a single selected path to their clients. While this helps keep path and session scale manageable, it makes BGP multipath unusable. We overcome this by using BGP ADD-PATH [RFC7911] between the RR and its clients (or among sub-ASes).

1.1.3. Summary

In summary, for a scale-out peering architecture:

- o BGP multipath needs to be enabled on all IBGP sessions inside the AS.

- o BGP multipath needs to be enabled on all EBGP sessions of each ASBR.

- o BGP ADD-PATH needs to be enabled on all IBGP sessions.

  - * RRs need to be able to send multiple paths per prefix. The upper limit depends on:

    - + The maximum number of ASBRs per site (say N).

    - + Possibly also on the maximum number of EBGP sessions held by a single ASBR with single peer AS (say M), depending on BGP next-hop attribute (BGP-NH) configuration.

  - * RR clients/ASBRs may need to be able to send multiple paths per prefix if BGP-NH configuration is "next hop unchanged". The upper limit depends on the maximum number of EBGP sessions held by a single ASBR with single peer AS (say M).

For further consideration the following network diagram will be used for reference:

```
   +-----------------------------------------------------------------+
   |    AS 1                                     +-------------------+|
   |  +-----------------------------------+      |+------+ SITE3 o--o ||
   |  | SITE1         +-------- Cost 10 -+------+|CR_3.1|--+  o-|RR| ||
   |  |   o------o    |                   |      |+------+   |  |Ro--o ||
   |  | O-|RR_1.1|    |                   |      |+------+   |  o--o   ||
   |  | |Ro------O    |          +--- Cost 10 --+|CR_3.K|   |+-------+||
   |  | O------O  +------+       |       |      |+---+--+   ||BR_3.N"||
   |  |           |CR_1.1|-------+- Cost 10 -+   |    |     |+-------+||
   |  |           +------+       |       |   |  +----+-----+---------+
   |  |           / / \      +------+    |   |    Cost 15 Cost 15     |
   |  |          / /   \     |CR_1.K|--Cost+ |  +----+-----+---------+
   |  |         /  |    \    +------+   10 | |  |+---+--+   |   SITE2 |
   |  |        /   |     \    /  | \     | | +--+|CR_2.K|   |  o--o   ||
   |  |       /    |      \--X-\ /  \    | |    |+------+   |  |RR|-o ||
   |  |      /  /--+--------/   X   |    | |    |+------+   |  o--oR| ||
   |  |     /  /   |   /-------/ \  \    | +----+|CR_2.1|<-+    o--o ||
   |  |    /  /    |  /           \  \   |      |+------+            ||
   |  | +------+ +------+       +------+ |      |+------+   +-------+||
   |  | |BR_1.1| |BR_1.2|-  -  -|BR_1.N| |      ||BR_2.1|   |BR_2.N'||
   |  | +X----X+ +-X---X+       +-X---X+ |      |+-+--+-+   +-+---+-+||
   |  +---X----X----X---X--------X-----X-+      +--+--+-------+---+--+|
   +-------X----X----X---X-------+------X----------+--+-------+---+---+
            \    \   |    \      |       \----\    |  |       |   |
    BR_1.1   \    \  |     \-----+----------\  \   |  |       \   |
     ^        \-\  \-+-----------+-------\   \  \  \  \        \  \
     X BR_1.2    \   |           |        \   \  \  \  \        \  \
     X  ^         \  |           /         \   \ |   \  \       |  |
     X  X BR_1.N   \ \   /------/           \  | |    \  \      |  |
     X  X  ^        \ \ /                    \ | |     \  \     |  |
     X  X  X        | | |   ^ ^ ^             | | |      \  \    |  |
     X  X  X        | | |   | | |             | | |       \  \   |  |
   +---------+ +----+-+-+---+-+-+-------------+-+-+--------X--X--+--+--+
   |         | |    | | |   | | |             | | |         \  \ |  |  |
   |         | |  +-+-+-++ ++-+-+-+        +------+       +------+ |  |
   |         | |  |PR_2.1| |PR_2.2|-  -  - |PR_2.M|       |PR_2.P+--+ |
   |         | |  +------+ +------+        +------+       +--+---+.T| |
   |         | |                                             +------+ |
   |   AS 3  | |                             AS 2                     |
   +---------+ +------------------------------------------------------+
   |=================================================================|
   |CR - Core Router                                                 |
   |BR - ASBR and/or Customer Edge in AS1                            |
   |PR - ASBR in peering ASes                                        |
   |=================================================================|
```

Figure 2

1.2. Common BGP Deployment Configurations

1.2.1. IBGP with Next-Hop Unchanged

In one standard BGP configuration, an ASBR, when it advertises an externally learned prefix into IBGP, does not modify the BGP-NH. So, the BGP-NH is set to the IP address of an interface on the external peering router. The strength of this technique is the shorter time needed to restore connectivity with all equal cost multi-path (ECMP) in-use and on low latency paths. The drawback is extremely high BGP Routing Information Base (RIB) scale - proportional to the number of inter-AS links.

1.2.1.1. Example

Let's assume that in the network of Figure 2, all PR2.x of AS2 advertise the same set of prefixes on all sessions to AS1.

If BR1.1-BR1.N and BR2.1-BR2.N' each advertise only one path per prefix to their respective RRs, then as the result of ADD-PATH among RRs, BRs and CRs, at site 3 the BRs and CRs will learn N+N' paths per prefix learned from AS2. This is sufficient to equally distribute load among all N ASBRs on site 1 (note the IGP cost between site 2 and site 3).

However, when interfaces over which all BR1.1-BR_1.N learned their best path become unavailable (say interfaces to PR_2.1 in all cases, as a result of the failure of PR_2.1), the route to the BGP BGP-NH - that is, the IP address of the PR_2.1 interface - is removed from the IGP. BGP speakers at other sites (BR_3.x) will react by temporarily directing traffic to site 2 (BR_2.1-BR_2.N'). This switchover may happen in sub-second time, in a prefix-scale-independent manner, thanks to techniques commonly known as BGP PIC Edge [I-D.ietf-rtgwg-bgp-pic]. As a result, traffic is on a path other than the lowest cost path, as the connection from site 1 to AS2 is not entirely broken (links to PR_2.2-PR_2.M are operational).

Subsequently, all BR1.x will update their RRs with a new best path (say for PR_2.2) for each prefix (for example, 100,000 of them), triggering global convergence. Such a convergence, for a large number of prefixes, may take many minutes.

In the above example, BRs, RRs, and possibly CRs keep N+N' paths per prefix (N from site 1, and N' from site 2). Provided N=N'=4, this makes 8 path per prefix.

The solution for sub-optimal routing right after the failure would be to enable each BR to advertise multiple paths to its RRs, and for

them in turn to propagate it to all other RRs and hence BRs. So, each of BR1.x at site 1 will advertise M paths (from PR_2.1-PR_2.M), RR1.x will have N*M ECMP best paths and advertise them to other sites (site 3). As a result, BGP speakers at other sites (BR3.x at site 3) are provided with N*M paths per prefix from site 1 and N'*M' from site 2. Therefore to achieve optimal routing immediately after failure, a considerably higher scale of BGP paths needs to be handled. If M=N=N'=M'=4 then for each prefix we have 16 best paths and 16 non-best, a total of 32. If AS2 advertises 100,000 prefixes, this becomes 3.2M paths.

Although this solution provides a mean of fast, prefix-scale-independent traffic switchover, it does it only if an ASBR external interface goes down, which triggers an IGP event. In case an EBGP session fails but the underlying interface remains up (misconfiguration, software defect, etc), recovery still requires per-prefix withdrawal/update that could take many minutes at high scale.

1.2.2. IBGP with Next-Hop-Self

The other common technique is to modify BGP-NH to "self" (a local IP address, typically a loopback) when the BR advertises an externally learned path into IBGP. This technique allows the reduction of the number of paths per prefix, while keeping optimal forwarding - least cost and ECMP - in case of failure discussed above (e.g. PR_2.1 node failure). Actually, because IP addresses of BGP-NH as seen by other BGP speakers do not change in response to external failure events, and are resolvable by the IGP, there is no need to reprogram the Forwarding Information Base (FIB) at all. Unfortunately, other failures - loss of all connectivity between a single BR (say BR1.1) and a peer AS (all PRs in AS2) would not be handled quickly. As the BGP-NH advertised by BR_1.1 is not changed and is reachable by the IGP, BGP speakers in AS1 (BRs, CRs) will keep BR_1.1 as a feasible exit point until they receive BGP withdraws on a prefix-by-prefix basis. This is a global convergence process that at high scale can take minutes, during which time packets may be discarded or loop.

2. The BGP Abstract Next-Hop

The Abstract Next Hop (ANH) concept presented below does not require any changes to the BGP protocol itself. It is architectural solution to network configuration, that uses existing protocols' capabilities while achieving higher scale and faster routing convergence when scale-out peering sites exist.

When a BGP speaker advertises a path to its IBGP peer, it modifies the Protocol Next-Hop to be the ANH value. The ANH is just an IP

address that identifies the BGP session or a set of BGP sessions. The set of BGP sessions is defined by the operator in local configuration, according to network design needs. For example, an ANH might identify:

- o a set of BGP sessions with the same peer AS and handled by a given single ASBR

- o a set of BGP sessions with same the peer AS and handled by one or more ASBRs at a given site

- o a set of BGP sessions with any upstream provider AS

- o a set of BGP sessions with a given peer device and handled by one or more of ASBRs of the local AS

A host route to the ANH is installed in the relevant RIB and redistributed into the IGP. BGP maintains the ANH host route based on the state of the associated group of BGP sessions:

- o As soon as all BGP sessions in the set go down, the ANH route is removed.

- o When at least one BGP session in of the set comes up, the ANH route is created only after initial route convergence is complete for the peer (End-of-RIB (EoR) [RFC4724] is received).

Taken together, these procedures ensure that as soon as the final session in the set goes down, ingress routers will see the associated ANH withdrawn from the IGP. Since the ANH is used to resolve the associated BGP next hops, the ingress routers are triggered to converge to send traffic to their alternate (new best) route. They also ensure that as soon as one session in the set comes up and is synchronized (that is, the EoR is received), ingress routers will see the ANH advertised in the IGP and will be able to reconverge to use routes that are associated with that next hop.

The ANH can be any IP address that the router is eligible to advertise according to the local network's IP address management scheme. More details are given in Section 3.3.

3. Use of Abstract Next-Hop in scale-out peering design

In traditional configurations as described in Section 1.2 the meaning of the BGP-NH is either:

- o An egress interface in the case of next-hop-unchanged configuration, or

o An egress ASBR in the case of next-hop-self configuration.

The meaning of Abstract Next Hop is more context-dependent. This document describes network configurations when the BGP-NH identifies:

a. An (egress ASBR, peer AS) pair. The ANH should be advertised into the IGP if, and only if, the given egress ASBR has at least one EBGP session in the ESTABLISHED state with the given peer AS, and the EoR marker has been received on that session. We call this the ASBR-Peer AS Abstract Next Hop (AP-ANH).

b. An (egress site in local AS, peer AS) pair, where a "site" may include multiple ASBRs. The ANH should be advertised into the IGP if, and only if, at least one ASBR of the given site has at least one EBGP session in the ESTABLISHED state with the given peer AS, and the EoR marker has been received on this session. We call this the Site-Peer AS Abstract Next Hop (SP-ANH).

Note that reachability of the ANH address in the IGP depends on EBGP session state and not inter-AS interface state, although of course, interface state may impact session state. How the IP route to the ANH address is instantiated on an ASBR and inserted into the IGP on particular device is a matter of local implementation.

3.1. Egress ASBR-Peer AS Abstract Next Hop (AP-ANH)

The AP-ANH is unique to an ASBR and its peer AS. For example, in the network of Figure 2, BR_1.1 would have two AP-ANH assigned - one for its peering with AS2 and the other for AS3. Similarly, BR_1.2 would have two AP-ANH, one per peer AS, with values different from the AP-ANH of BR_1.1, and so on. All AP-ANH are exported into the IGP by their ASBRs. Each ASBR advertises only one path per prefix to its RR, with the BGP-NH set to the appropriate AP-ANH. The RR will propagate it through the entire AS by means of IBGP ADD-PATH. In consequence, the number of paths learned per prefix is equal to number of ASBRs servicing a given peer AS. In the network as of Figure 2, for AS2 prefixes, this would be N+N' (from site_1 + from site_2) paths per prefix. This sets the scale requirements of this solution to be on par with Next-Hop-Self (Section 1.2.2). However, thanks to the properties of ANH, more failures are covered by prefix-independent techniques, as withdrawal of the ANH from the IGP makes the BGP-NH unresolvable.

Provided that all ASBRs in a given site (site1 in Figure 2) receive the same routing information from their peer AS (AS2), in non-faulty conditions, one could consider setting the ANH value on all ASBRs the same. However, failure(s) can create situations when multiple ASBRs will have a session in ESTABLISHED state with a given peer AS, but

some prefixes would be learned from EBGP only on a subset of these ASBRs. To prevent problems from arising in this situation, the per-ASBR AP-ANH needs to be advertised into the IGP and ASBRs need to set it as the BGP-NH when advertising routes to the site's Route Reflectors. However, for IBGP path advertisement being propagated beyond the site (into the RR mesh), the BGP-NH may be replaced by another ANH value, the Site-Peer AS ANH.

3.2. The Site-Peer AS Abstract Next Hop (SP-ANH)

The AP-ANH works on an ASBR level. From a given local AS perspective, the number of ANH is proportional to the number of pairs of ASBRs and ASes each of them peers with. With hundreds of peer ASes, tens of sites and ~10 ASBRs per site, the number of AP-ANH may scale into the thousands. At the same time, it may not be necessary or even desirable for every BGP speaker in the network to have visibility to every path down to individual egress ASBR granularity. With symmetrical multiplane backbone and/or leaf-spine designs, it is sufficient that BGP speakers on other sites have information that a given site (site1 in Figure 2) has at least one ASBR with an ESTABLISHED session to the peer AS (AS2). For example, in the network of Figure 2, even if BR3.1 has only one path with its BGP-NH equal to the ANH of BR1.1, BR3.1 resolves the BGP-NH in the IGP and spreads traffic among all CRs on site 3. Thus, traffic will be delivered to CR1.x at site 1. As long as CR1.x has visibility to all paths, traffic will be distributed equally to all site 1 ASBRs.

At the same time, when multiple paths are available on BGP speakers, every change is propagated, with consequent transmission and processing costs on all BGP speakers across the network. This will be true even if the route change doesn't impact the forwarding plane. For example, in the network of Figure 2, even if BR3.1 has N paths with BGP-NHs set to the ANHs of BR1.1 through BR1.N, BR3.1 will resolve those BGP-NHs in the IGP and spread traffic among all CRs of site 3. When one of the egress ASBRs (say BR1.2) loses its connectivity to the peer AS, the affected BGP routes (those with BGP-NH equal to AP-ANH of BR1.2) are withdrawn from all BGP speakers (e.g. BR3.1) of the network. All BGP speakers perform path selection and possibly update their forwarding data structures. Since the actual forwarding paths do not change, all this work represents unnecessary churn.

To avoid the above drawbacks, the RR of a given site (site1 in Figure 2), when re-advertising a BGP path learned from its ASBR client, modifies the BGP-NH to another abstract value - the Site-Peer AS Abstract NH (SP-ANH). This value is unique per (site, peer AS) pair, and is shared by all RRs of a given site. With this modification, it is sufficient that inter-site IBGP sessions carry

only one path per prefix (no ADD-PATH needed). Consequently, BGP RIB scale is reduced significantly. This frees up memory, reduces the amount of data RRs need to exchange, and mitigates churn. The BGP speakers in other sites of AS 1 need to resolve SP-ANH in order to build their local FIBs. Therefore SP-ANH have to be present in the IGP - some router(s) in the local site (RR, ASBR or CR) need to inject it into the IGP. While the selection of role that is responsible of SP-ANH injection is discussed below, in any case, the SP-ANH should be reachable in the IGP if, and only if, at least one of AP-ANH (for the same peer AS and ASBR belonging to given site) is reachable. Figure 3 illustrates routing information flow in a network such as that of Figure 2:

```
                        +-----------------------------------------------
                        |                            +----->IBGP to SITE2
                        |                   AS 1     | +--->IBGP to SITE3
   /============================\                     | |
   |a.a.a.a/a                   |----------------->| |   SP-ANH
   |  as-path   "^2 .*"         |                  | |       (SITE1&AS2)
   |  BGP-NH    SP-ANH(SITE1&AS2)|                  | |   IP/32 into IGP
   \============================/                   | |             ^
                        |                            | |             |
                        |  +-------------------------+-+------------+---+
   /==============================\    o------o   o-+-+--o             |
   |ADD-PATH                      |    |RR_1.2|   |RR_1.1|  SITE1      |
   |a.a.a.a/a                     |    o------O   o----X-O             |
   |  as-path   "^2 .*"           |                ^ ^  \              |
   |  BGP-NH    AP-ANH(BR_1.1&AS2)|               / /    \              |
   |a.a.a.a/a                     |--------------X-X---->|             |
   |  as-path   "^2 .*"           |             /  |     |             |
   |  BGP-NH    AP-ANH(BR_1.2&AS2)|            /   |     |             |
   \==============================/           /    |     |             |
   /==============================\          /     |     \             |
   |a.a.a.a/a                     |          |     |      \            |
   |  as-path   "^2 .*"           |---------->/    |       v           |
   |  BGP-NH    AP-ANH(BR_1.1&AS2)|         /      |   +------+        |
   \==============================/        /       |   |CR_1.1+--+     |
   /==============================\       /        /   +--+---+.1+-+    |
   |a.a.a.a/a                     |------X------->/       +-+----+X|   |
   |  as-path   "^2 .*"           |     /        /          +------+   |
   |  BGP-NH    AP-ANH(BR_1.2&AS2)| +------+ +------+       +------+   |
   \==============================/ |BR_1.1| |BR_1.2|-  -  -|BR_1.N|   |
                        |  |        +------+ +------+       +------+   |
                        |  |          ^  ^                             |
                        |  |           \  \                            |
                        |  +-------------X--X---------------------------+
   /======================\--------------X--X---------------------------
   |a.a.a.a/a             |               \  \
   |  as-path   "^2 .*"   |--------------->\  \---------\
   \======================/                 \            \
   /======================\                  \            \
   |a.a.a.a/a             |-------------------X----------->\
   |  as-path   "^2 .*"   |----------------+ +-X------------X-----------
   \======================/                | | +X-----+   +--X---+     +
                        |        AS 3     | | |PR_2.1|   |PR_2.2|- - -|
                        |                 | | +------+   +------+     +
                        |                 | |        AS 2
                        +-------------------+ +----a.a.a.a/a network-----
```

Figure 3

3.3. Assignment of Abstract Next Hops

In the following subsections we provide more details of how abstract next hops can be injected in several different common network architectures.

3.3.1. Native IP Networks

In this network every router, including core routers, has full BGP routing information and forwards each packet based on destination IP lookup. Provided that all routers at an egress site receive multiple paths with BGP-NH set to AP-ANH (and not SP-ANH), it is a matter of the operator's decision which node - RR, ASBR or CR - will inject the SP-ANH route into the IGP. One may argue that injection of SP-ANH by ASBRs may be simpler, as it will be done by the same procedure and policy as injection of AP-ANH. Others may prefer injection at RR, as it limits the number of configuration touch-points.

3.3.2. MPLS

3.3.2.1. Identical BGP address space and paths received on all ASBRs

In the MPLS network, since traffic is carried over LSP tunnels, the SP-ANH needs to be injected into the IGP by a node that has the ability to perform an IP lookup. This eliminates the RR, and possibly CRs (in "BGP-free core" architectures). Instead, all ASBRs are used to insert SP-ANH addresses into the IGP. In case of LDP-based networks, this is sufficient. The CR will create an ECMP forwarding structure for labels of SP-ANH FEC coming from other sites. In RSVP-TE based networks, ECMP needs to happen on the ingress LSR and therefore, every BGP speaker needs to establish an LSP to every ASBR, and the SP-ANH address needs to be part of the FEC for its respective LSP. If SP-ANH is used as an RSVP (signaling) destination, some other means (such as affinity groups) needs to be used to ensure the desired 1:1 LSP to egress ASBR mapping.

3.3.2.2. Different address space sets or paths received on different ASBRs

In the case when the set of prefixes received from a given peer AS by one ASBR is different from the set received by another one, a combination of SP-ANH and MPLS-based load balancing on a CR may lead to a situation where an IP packet will be directed to an ASBR that lacks external routing information and hence can't forward traffic directly out of the AS. Similarly, if path attributes for a given prefix received by one ASBR are different from those received by another, again packets can be directed to the "wrong" ASBR. In this case the ASBR would use the IBGP route it learned from another ASBR

of the same site (via RR, with AP-ANH) and forward traffic over an LSP to the "correct" ASBR. This extra hop constitutes a sub-optimal traffic path through the network.

For example in the network of Figure 2, let's assume that prefix P2 is advertised to BR1.2-BR1.N by AS2 but not to BR1.1. BR3.1 has a BGP best route to P2 with its BGP-NH set to the SP-ANH of (site1, AS2). It resolves it by ECMP over N MPLS LSPs, terminating on BR1.1-BR1.N. So, some packets are forwarded by BR3.1 over an LSP via CR1.x and terminated on BR1.1. BR1.1 has no external route to P2, but it has (N-1) IBGP routes to P2 w/ BGP-NHs equal to the AP-ANHs of BR1.2-BR1.N. Therefore BR1.1 performs an IP lookup and forwards this packet over LSPs via CR1.x and terminated on BR1.2-BR1.N. Traffic is U-turned on BR1.1 and traverses CRs at site 1 twice.

Such asymmetry may be considered acceptable by the provider, as long as it's a transient condition. However, in the general case such a situation could be persistent, as the result of intentional configuration on the peer AS's ASBRs. Therefore the better solution would be to insert the SP-ANH into the IGP on CRs. In this case, CRs need to perform forwarding based on destination IP lookup. Therefore CRs would have to be able to learn and handle large IP routing and forwarding tables - at least all prefixes learned from peer ASes by the local ASBRs.

3.3.3. SPRING

3.3.3.1. Identical BGP address space and path received on all ASBRs

For SPRING based networks, we can take advantage of the unique capability of Anycast-SID [RFC8402]. The ASBRs of a single site allocate an Anycast-SID for each SP-ANH address. This SID can be used as the only SID by an ingress BGP speaker or, if a TE routed path is desired, depending on TE constraints, the TE controller can provision a SPRING path with the Anycast-SID at the end, instructing the CR to perform load balancing among connected ASBRs.

3.3.3.2. Different address space sets or paths received on different ASBRs

Similarly to a classic MPLS environment, such a situation may lead to suboptimal routing (redirecting from one ASBR to another), or may require the CR (instead of ASBR) to insert the SP-ANH into the IGP and generate a PREFIX-SID (or Anycast-SID if there is more then one CR) for it.

4. Worked Examples

Below we illustrate the operation of the proposal by working through its operation in the context of several different types of failures. Here, we assume that each ASBR in a given site of the local AS (site 1 of AS1 in Figure 2), that has an EBGP session with the given peer AS (AS2 in Figure 2), receives from its peer routers (PR2.x) routes to exactly same address space on each session.

4.1. Failure of a proper subset of EBGP sessions with a given peer AS on a single ASBR

- The impacted ASBR keeps advertising the AP-ANH into the IGP, as at least one session to the peer AS remains in the ESTABLISHED state.
- The impacted ASBR may send UPDATEs to RRs, however the BGP-NH remains the same and equal to the pre-failure AP-ANH.
- The RRs may send UPDATEs to their clients (CRs, BRs) and to RRs in other sites, however the BGP-NH remains the same as its pre-failure value: AP-ANH and SP-ANH respectively.
- As BGP-NH do not change, there are no changes in forwarding data structures (FIB) on any BGP speaker across the network, except possibly the ASBR that holds the impacted session.

4.2. Failure of a proper subset of EBGP sessions with a given peer AS on each ASBR of a given site

- The impacted ASBRs keep advertising the AP-ANH into the IGP, as at least one session to the peer AS remains in the ESTABLISHED state on each ASBR.
- The impacted ASBRs may send UPDATEs to RRs, however the BGP-NH remains the same and equal to the pre-failure AP-ANH.
- The RRs may send UPDATEs to their clients (CRs, BRs) and to RRs in other sites, however the BGP-NH remains the same and equal to its pre-failure value: AP-ANH and SP-ANH respectively.
- As BGP-NH do not change, there are no changes in forwarding data structures (FIB) on any BGP speaker across the network, except possibly the ASBRs that hold the impacted sessions.

4.3. Failure of all EBGP sessions with a given peer AS on single ASBR; Failure of a single ASBR

- The impacted ASBR stops advertising the AP-ANH into the IGP, as it has lost all sessions with given peer AS.

- The SP-ANH is kept reachable in the IGP.

- All other BGP speakers at the impacted site invalidate all paths with BGP-NH equal to the AP-ANH. This may trigger prefix-independent FIB data-structure patching/temporary fixing for sub-second traffic restoration.

- The impacted ASBR sends WITHDRAWs to its RRs.

- Each RR:

  * Sends WITHDRAWs to its clients at the local site (CRs, BRs) for paths from the impacted ASBR. As these sessions support ADD-PATH, paths from other ASBRs will remain. Other BGP speakers at this site have to modify their FIBs.

  * May send UPDATEs to RRs in other sites, however the BGP-NH remains the same, equal to the pre-failure SP-ANH. As the BGP-NH does not change, there are no changes in forwarding data structure (FIB) on any of BGP speakers across network, except those at the impacted site.

- Routing churn is mitigated in many cases to a single peering site, and does not propagate across the network. FIB changes are limited to a single peering site, and do not propagate across the network.

4.4. All EBGP sessions with a given peer AS on all ASBRs

- Each ASBR stops advertising its AP-ANH into the IGP, as it has lost all sessions with the given peer AS.

- The SP-ANH is no longer reachable in the IGP, as none of AP-ANH are reachable.

- All other BGP speakers across the network invalidate all paths with a BGP-NH equal to the removed AP-ANH or SP-ANH. This may trigger prefix-independent FIB data-structure patching/temporary fixing for sub-second traffic restoration.

- Each impacted ASBR sends WITHDRAWs to its RRs.

- o The RRs send WITHDRAWs to their clients at the local site (CRs, BRs) and RRs in other sites for paths from the impacted ASBRs. As these sessions support ADD-PATH, paths from ASBRs at other sites will remain. The BGP speakers across the network may need to modify their FIBs.

5. Acknowledgements

Valuable comments and suggestions on solution covered by this document was provided by Mannan Venkatesan, John Scudder and Ron Bonica. Special thanks to John Scudder, who also helped with editorial changes.

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

Since this is a deployment architecture and not a protocol modification, it doesn't introduce any new issues to the BGP protocol itself. General BGP security considerations are discussed in [RFC4271] and [RFC4272], BGP deployment best practices are documented in [RFC7454], and nothing in this proposal impedes their use. Many of the practices recommended in that document are self-evidently still applicable, for example the use of cryptographic session protection methods such as TCP MD5 [RFC2385] or the TCP Authentication Option [RFC5925], and the Generalized TTL Security Mechanism [RFC5082]. Since we propose a novel use of IP addresses to assign ANHs, it's worth considering if anything new is required to protect them. We conclude there isn't, they fall into the existing category of "Prefixes Belonging to the Local AS" discussed in section 6.1.4 of [RFC7454].

8. Informative References

[I-D.ietf-rtgwg-bgp-pic]
Bashandy, A., Filsfils, C., and P. Mohapatra, "BGP Prefix Independent Convergence", draft-ietf-rtgwg-bgp-pic-08 (work in progress), September 2018.

[RFC2385] Heffernan, A., "Protection of BGP Sessions via the TCP MD5 Signature Option", RFC 2385, DOI 10.17487/RFC2385, August 1998, <https://www.rfc-editor.org/info/rfc2385>.

[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>.

[RFC4272] Murphy, S., "BGP Security Vulnerabilities Analysis", RFC 4272, DOI 10.17487/RFC4272, January 2006, <https://www.rfc-editor.org/info/rfc4272>.

[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, <https://www.rfc-editor.org/info/rfc4456>.

[RFC4724] Sangli, S., Chen, E., Fernando, R., Scudder, J., and Y. Rekhter, "Graceful Restart Mechanism for BGP", RFC 4724, DOI 10.17487/RFC4724, January 2007, <https://www.rfc-editor.org/info/rfc4724>.

[RFC5065] Traina, P., McPherson, D., and J. Scudder, "Autonomous System Confederations for BGP", RFC 5065, DOI 10.17487/RFC5065, August 2007, <https://www.rfc-editor.org/info/rfc5065>.

[RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., Ed., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, DOI 10.17487/RFC5082, October 2007, <https://www.rfc-editor.org/info/rfc5082>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <https://www.rfc-editor.org/info/rfc5925>.

[RFC7454] Durand, J., Pepelnjak, I., and G. Doering, "BGP Operations and Security", BCP 194, RFC 7454, DOI 10.17487/RFC7454, February 2015, <https://www.rfc-editor.org/info/rfc7454>.

[RFC7911] Walton, D., Retana, A., Chen, E., and J. Scudder, "Advertisement of Multiple Paths in BGP", RFC 7911, DOI 10.17487/RFC7911, July 2016, <https://www.rfc-editor.org/info/rfc7911>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

Authors' Addresses

Rafal Jan Szarecki (editor)
Juniper Networks Inc.
1133 Innovation Way
Sunnyvale, CA 94089
US

Phone: +1(408)680-9604
Email: rafal@juniper.net

Kaliraj Vairavakkalai
Juniper Networks Inc.
1133 Innovation Way
Sunnyvale, CA 94089
US

Phone: +1(408)936-8872
Email: kaliraj@juniper.net

Natrajan Venkataraman
Juniper Networks Inc.
1133 Innovation Way
Sunnyvale, CA 94089
US

Phone: +1(408)936-6597
Email: natv@juniper.net

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 22 April 2024

F. Xu
Tencent
T. Graf
Swisscom
Y. Gu
S. Zhuang
Z. Li
Huawei

20 October 2023

BGP Route Policy and Attribute Trace Using BMP
draft-xu-grow-bmp-route-policy-attr-trace-08

Abstract

The generation of BGP adj-rib-in, local-rib or adj-rib-out comes from BGP route exchange and route policy processing. BGP Monitoring Protocol (BMP) provides the monitoring of BGP adj-rib-in [RFC7854], BGP local-rib [RFC9069] and BGP adj-rib-out [RFC8671]. By monitoring these BGP RIB's the full state of the network is visible, but how route-policies affect the route propagation or changes BGP attributes is still not. This document describes a method of using BMP to record the trace data on how BGP routes are (not) changed under the process of route policies.

Requirements Language

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 RFC 2119 [RFC2119].

Status of This Memo

Table of Contents

1. Introduction

The typical processing procedure after receiving a BGP Update Message at a routing device is as follows: 1. Adding the pre-policy routes into the pre-policy adj-rib-in (if any); 2. Filtering the pre-policy routes through inbound route policies; 3. Selecting the BGP best routes from the post-policy routes; 4. Adding the selected routes into the BGP local-rib; 5-a. Adding the BGP best routes from local-rib to the core routing table manager for selection; 5-b. Filtering the routes from BGP local-rib through outbound route policies w.r.t. per peer or peer groups; 6. Sending the BGP adj-rib-out to the target peer or peer groups. Details may vary by vendors. The BGP Monitoring Protocol (BMP) can be utilized to monitor BGP routes in

forms of adj-rib-in, local-rib and adj-rib-out. However, the complete procedure from inbound to outbound policy processing, including other policies, e.g., route redistribution, route selection and so on, is currently unobserved. For example, there are 10 policy items (or nodes) configured under one outbound route policy per a specific peer. By collecting the local-rib and adj-rib-out through BMP, the operator finds that the outbound policy didn't work as expected. However, it's hard to distinguish which one of the 10 policy items/nodes is responsible for the failure.

1.1. BGP Route Policy and Attribute Trace Overview

This document describes a method that records and reports how each policy item/node processes the routes (e.g., changes the route attribute). Each policy item/node processing is called an event thereafter in this document. Compared with conventional BGP rib entry, which consists of prefix/mask, route attributes, e.g., next hop, MED, local preference, AS path, and so on, the event record discussed in this document includes extra information, such as event index, timestamp, policy information, and so on. For example, if a route is processed by 5 policy items/nodes, there can be 5 event records for the same prefix/mask. Each event is numbered in order of time (e.g., the time of policy execution). The policy information includes the policy name and item/node ID/name so that the server/controller can map to the exact policy either directly from the device or from the configurations collected at the server side.

This document defines a new BMP message type to carry the recorded policy and route data. More detailed message format is defined in Section 2. The message is called the BMP Route Policy and Attribute Trace Message thereafter in this document.

1.2. Use cases

There are cases that a new policy is configured incorrectly, e.g., setting an incorrect community value, or policy placed in incorrect order among other policies. These may result in incorrect route attribute modification, best route selection mistake, or route distribution mistake. With the correlated record of policy and route, the server/controller is able to identify the unexpected route change and its responsible policy. Considering the fact that the BGP route policy impacts not only the route processing within the individual device but also the route distribution to its peers, the route trace data of a single device is always analyzed in correlation with such data collected from its peer devices.

Apart from the policy validation application, the route trace data can also be analyzed to discover the route propagation path within the network. With the route's inbound and outbound event records collect from each related device, the server is able to find the propagation path hop by hop. The identified path is helpful for operators to better understand its network, and thus benefiting both network troubleshooting and network planning.

2. Extension of BMP for Route Policy and Attribute Trace

2.1. Common Header

This document defines a new BMP message type to carry the Route Policy and Attribute Trace data.

* Type = TBD: Route Policy and Attribute Trace Message

The new defined message type is indicated in the Message Type field of the BMP common header.

2.2. Per Peer Header

The Route Policy and Attribute Trace Message is not per peer based, thus it does not require the Per Peer Header.

2.3. Route Policy and Attribute Trace Message

The Route Policy and Attribute Trace Message format is defined as follows:

```
+-------------------------------------------------------------+
|V|                          Reserved                         |
+-------------------------------------------------------------+
|                       Route Distinguisher                   |
+-------------------------------------------------------------+
|                         Prefix length                       |
+-------------------------------------------------------------+
|                             Prefix                          |
+-------------------------------------------------------------+
|                          Route Origin                       |
+-------------------------------------------------------------+
|                         Event count                         |
+-------------------------------------------------------------+
|                      Total event length                     |
+-------------------------------------------------------------+
|                           1st Event                         |
+-------------------------------------------------------------+
|                           2nd Event                         |
+-------------------------------------------------------------+
~                                                             ~
+                            ......                           +
~                                                             ~
+-------------------------------------------------------------+
|                          Last Event                         |
+-------------------------------------------------------------+
```

Figure 1: Route Policy and Attribute Trace Message format

* Flags (1 Byte): The V flag indicates that the Peer address is an IPv6 address. For IPv4 peers, this is set to 0.

* Route Distinguisher (8 Bytes): indicates the route distinguisher (RD) related to the route.

* Prefix Length (1 Byte): indicates the length of the prefix.

* Prefix (16 Bytes): indicates the monitored prefix, with mask defined by Prefix Length field. It is 4 bytes long if an IPv4 address is carried in this field (with the 12 most significant bytes zero-filled) and 16 bytes long if an IPv6 address is carried in this field.

* Route Origin (4 Bytes): indicates the BGP router ID where this route is learned from. If the route is locally generated, this field is zero filled.

* Event Count (1 Byte): indicates the total number of policy processing event recorded in this message.

* Total event length (2 Bytes): indicates the total length of the following fields including all events, where the total number is indicated by the Event Count field.

* 1 ~ Last event: indicates each event, stacked one by one in order of time. The event format is further defined as follows.

```
+-------------------------------------------------------------+
|                      Single event length                    |
+-------------------------------------------------------------+
|                          Event index                        |
+-------------------------------------------------------------+
|                      Timestamp(seconds)                     |
+-------------------------------------------------------------+
|                      Timestamp(microseconds)                |
+-------------------------------------------------------------+
|                       Path Identifier                       |
+-------------------------------------------------------------+
|                             AFI                             |
+-------------------------------------------------------------+
|                             SAFI                            |
+-------------------------------------------------------------+
|                         VRF/Table TLV                       |
+-------------------------------------------------------------+
|                           Policy TLV                        |
+-------------------------------------------------------------+
|                    Pre Policy Attribute TLV                 |
+-------------------------------------------------------------+
|                    Post Policy Attribute TLV                |
+-------------------------------------------------------------+
|                           String TLV                        |
+-------------------------------------------------------------+
```

Figure 2: Event format

* Single event length (2 Bytes): indicates the total length of a single policy process event, including the following fields that belong to this event.

* Event index (1 Byte): indicates the sequence number of this event, staring from 1 and increases by 1 for each event recorded in order.

* Timestamp (8 Bytes): indicates the time when the policy of this event starts execution, expressed in seconds and microseconds since midnight (zero hour), January 1, 1970 (UTC).

* Path Identifier (4 Bytes): used to distinguish multiple BGP paths for the same prefix. If there's no path ID, this field is zero filled.

* AFI (2 Bytes)/SAFI (1 Byte): indicates the AFI/SAFI of the route.

* VRF/Table TLV (Variable): indicates the VRFinformation of the route. The format of the VRF/Table TLV is further defined in Figure 3. The VRF/Table ID TLV is optional. At most one VRF/Table TLV can be included in each Route Policy and Attribute Trace Message.

* Policy TLV (Variable): indicates the ID of the route policy of this event, which is user specific or vendor specific, which can be used for mapping to the actual policy content. The policy content data retrieval is out of the scope of this document. The format of the Policy ID TLV is further defined in Figure 4. The Policy ID TLV is optional. At most one Policy TLV can be included in each Route Policy and Attribute Trace Message.

* Pre Policy Attribute TLV (Variable): include the BGP route attributes before the policy is executed. The format of the Pre-policy Attribute TLV is further defined in Figure 4. The Pre-policy Attribute TLV is optional. At most one Pre Policy Attribute TLV can be included in each Route Policy and Attribute Trace Message.

* Post Policy Attribute TLV (Variable): include the BGP route attributes after the policy is executed. The format of the Post-policy Attribute TLV is further defined in Figure 5. The Post-policy Attribute TLV is optional. At most one Post Policy Attribute TLV can be included in each Route Policy and Attribute Trace Message.

* String TLV (Variable): leaves for future extension. The String TLV is optional. One or more String TLVs can be included in each Route Policy and Attribute Trace Message.

2.3.1. VRF/Table TLV

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-------------------------------+-------------------------------+
|          Type = TBD1          |            Length             |
+-------------------------------+-------------------------------+
|                         VRF/Table ID                          |
+---------------------------------------------------------------+
~                         VRF/Table Name                        ~
|                                                               |
+---------------------------------------------------------------+
```

Figure 3: VRF/Table TLV

* Type = TBD1 (2 Bytes): VRF/Table TLV.

* Length (2 Bytes): indicates the total length of the VRF/Table ID field and the VRF/Table Name field.

* VRF/Table ID (4 Bytes): indicates the VRF or table ID of this route.

* VRF/Table name (Variable): indicates the VRF or table name of this route in the format of ASCII string. The string size MUST be within the range of 1 to 255 bytes.

2.3.2. Policy TLV

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-------------------------------+-------------------------------+
|          Type = TBD2          |            Length             |
+---------------+-------------------------------+---------------+
|M|P|D|  Res.   |  Policy Count | Policy Class. |
+---------------+---------------+---------------+---------------+
~                         Peer Address                          ~
+                                                               +
+---------------------------------------------------------------+
|                         Peer Router ID                        |
+---------------------------------------------------------------+
|                           Peer AS                             |
+---------------------------------------------------------------+
~                                                               ~
+                          1st Policy          +---------------+
|                                              |C|R|   Res.    |
+---------------------------------------------------------------+
~                               .                               ~
+                               .                               +
~                               .                               ~
+---------------------------------------------------------------+
~                                                               ~
+                          Last Policy         +---------------+
|                                              |C|R|   Res.    |
+---------------------------------------------------------------+
```

Figure 4: Policy TLV

* Type = TBD2 (2 Bytes): Policy TLV.

* Length (2 Bytes): indicates the length of the Policy Value field that follows it. The Policy value field includes the reserved Flag Byte, Policy Count field, Policy Classification field, Peer Router ID field, Peer AS field, and each Policy field.

* Flag Byte (1 Byte): the M bit (the left most bit) indicates if the route in this event is matched (once or multiple times) or not by any policies. "0" means no match and "1" means else wise. When the M bit is set to "0", the Post Policy Attribute TLV SHALL not be included in the Message. The P bit (the second left bit) indicates if the matched result is Permit or Deny. "0" means Deny, and "1" means Permit. When the M bit is set to "0", any value of the P bit SHOULD be ignored. When the P bit is set to "0", the Post Policy Attribute TLV SHALL not be included in the Message. The D bit (the third left bit) indicates if there exists any difference between the pre-policy attributes and the post policy attributes. "0" means no difference, and "1" means difference exists. When the D bit is set to "0", the Post Policy Attribute TLV SHALL not be included in the Message.

* Policy Count (1 Byte): indicates the number of policies carried in this event.

* Policy Classification (1 Byte): indicates the category of the policy. Currently 8 policy categories are defined: "00000000" indicates the Inbound policy; "00000001" indicates the Outbound policy; "00000010" indicating the Multi-protocol Redistribute policy (including routes import from other protocols, like ISIS/ OSPF and static routes), "00000011" indicates the Cross-VRF Redistribute policy (route import between VRF and global table and between VRFs); "00000100" indicates VRF Import policy (e.g., an IPv4 route within a VRF transformed from a VPNv4 route), "00000101" indicates VRF Export policy (e.g., a VPNv4 route transformed from an IPv4 route within an VRF); "00000110" indicates the Network policy (BGP network installment and advertisement), "00000111" indicates the Aggregation policy; "00001000" indicating the Route Withdraw (triggered by BGP Update or local actions, e.g., route aggregation). Specifications regarding each category can be included in the String TLV. For the route update, i.e., route creation and withdrawal, that is not processed by any route policy, the Policy Category field is set per the route update point. In addition, the Policy ID field in the Policy ID TLV SHOULD be set to 0.

Value	Policy Classification
00000000	Inbound policy
00000001	Outbound policy
00000010	Multi-protocol Redistribute
00000011	Cross-VRF Redistribute
00000100	VRF import
00000101	VRF export
00000110	Network
00000111	Aggregation
00001000	Route Withdraw

Table 1: Policy Classification

* Peer Address: The remote IP address associated with the TCP session over which the encapsulated PDU was received. It is 4 bytes long if an IPv4 address is carried in this field (with the 12 most significant bytes zero-filled) and 16 bytes long if an IPv6 address is carried in this field.

* Peer Router ID (4 Bytes): indicates the BGP Router ID where this policy is configured under. This field is used in combination with the Policy Classification field. If the Policy Classification field is set to "00000000", meaning Inbound policy, then this field is set to the BGP router ID where the route is received from; if the Policy Classification field is set to "00000001", meaning Outbound policy, then this field is set to the BGP router ID where the route is distributed to; If the Policy Direction field is set to any other values, then this field is set to all zeros.

* Peer AS (4 Bytes): indicates the AS number of the BGP Peer that defined the Peer ID field.

* 1st ~ Last Policy (Variable): indicates the Policy name and the Item ID of each policy match.

* Flag Byte (1 Byte): the C bit (left most bit) indicates if the next subsequent policy has chaining relationship to the current policy. "1" means it’s chaining relationship and "0" means else wise. For the flag byte following the Last Policy field, the C bit SHALL be set to "0". The R bit (second left bit) indicates if the next subsequent policy has recursion to the current policy. "1" means it’s recursion and "0" means else wise. For the flag byte following the Last Policy field, the R bit SHALL be set to "0".

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-------------------------------+-------------------------------+
|       Policy Name Length      |      Policy Item ID Length    |
+-------------------------------+-------------------------------+
~                           Policy Name                         ~
+                                                               +
+---------------------------------------------------------------+
~                         Policy Item ID                        ~
+                                                               +
+---------------------------------------------------------------+
```

Figure 5: Policy field format

The Policy field consists of the Policy Name (Variable) and the Policy Item ID (Variable). The Policy Name and Policy Item ID fields are in the format of ASCII string. The length of Policy Name is indicated by the Policy Name Length (2 Bytes) field. The length of Policy Item ID is indicated by the Policy Item ID Length (2 Bytes) field.

2.3.3. Pre Policy Attribute TLV

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-------------------------------+-------------------------------+
|          Type = TBD3          |            Length             |
+-------------------------------+-------------------------------+
~                  Pre Policy Attribute sub TLVs                ~
+                                                               +
+---------------------------------------------------------------+
```

Figure 6: Pre Policy Attribute TLV

* Type = TBD3 (2 Bytes): Pre Policy Attribute TLV.

* Pre Policy Attribute length (2 Bytes): indicates the total length of the following Pre Policy Attribute sub TLVs.

* Pre Policy Attribute sub TLVs (Variable): include the BGP route attributes before the policy is executed.

2.3.4. Post Policy Attribute TLV

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-------------------------------+-------------------------------+
|          Type = TBD4          |            Length             |
+-------------------------------+-------------------------------+
~                 Post Policy Attribute sub TLVs                ~
+                                                               +
+---------------------------------------------------------------+
```

Figure 7: Post Policy Attribute TLV

* Type = TBD4 (2 Bytes): Post Policy Attribute TLV.

* Post Policy Attribute length (2 Bytes): indicates the total length of the following Post Policy Attribute sub TLVs.

* Post Policy Attribute sub TLVs (Variable): include the BGP route attributes after the policy is executed.

2.3.5. String TLV

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-------------------------------+-------------------------------+
|          Type = TBD5          |            Length             |
+-------------------------------+-------------------------------+
~                             Value                             ~
+                                                               +
+---------------------------------------------------------------+
```

Figure 8: String TLV

* Type = TBD5 (2 Bytes): String TLV.

* Length (2 Bytes): indicates the length of the following value field.

* Value (Variable): the textual expression of user-specific information in ASCII format.

An example of using the String TLV is expressing the route policy xpath information instead of using the Policy TLV.

3. Implementation Considerations

Considering the data amount of monitoring the route and policy trace of all routes from all BMP clients, users MAY trigger the monitoring at any user-specific time. Users MAY configure locally at the BMP client to monitor only user-specific routes or all the routes. In addition, users MAY configure locally at the BMP client whether to report the TLVs that are optional according to their own requirements, i.e., the Pre Policy Attribute TLV, Post Policy Attribute TLV, Policy ID TLV, and String TLV.

Successive recored events from one device MAY be encapsulated in one Route Policy and Attribute Trace Message or multiple Route Policy and Attribute Trace Messages per the user configuration.

4. Acknowledgments

TBD.

5. IANA Considerations

This document defines the following new BMP Message type (Section 2.1).

* Type = TBD: Route Policy and Attribute Trace Message.

This document defines the following new TLV types for the Route Policy and Attribute Trace Message (Section 2.3).

* Type = TBD1 (2 Bytes): VRF/Table TLV.

* Type = TBD2 (2 Bytes): Policy TLV.

* Type = TBD3 (2 Bytes): Pre Policy Attribute TLV.

* Type = TBD4 (2 Bytes): Post Policy Attribute TLV.

* Type = TBD5 (2 Bytes): String TLV.

6. Security Considerations

TBD.

7. 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>.

[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>.

[RFC5492] Scudder, J. and R. Chandra, "Capabilities Advertisement with BGP-4", RFC 5492, DOI 10.17487/RFC5492, February 2009, <https://www.rfc-editor.org/info/rfc5492>.

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/info/rfc7854>.

[RFC8671] Evens, T., Bayraktar, S., Lucente, P., Mi, P., and S. Zhuang, "Support for Adj-RIB-Out in the BGP Monitoring Protocol (BMP)", RFC 8671, DOI 10.17487/RFC8671, November 2019, <https://www.rfc-editor.org/info/rfc8671>.

[RFC9069] Evens, T., Bayraktar, S., Bhardwaj, M., and P. Lucente, "Support for Local RIB in the BGP Monitoring Protocol (BMP)", RFC 9069, DOI 10.17487/RFC9069, February 2022, <https://www.rfc-editor.org/info/rfc9069>.

Authors' Addresses

Feng Xu
Tencent
Guangzhou
China
Email: oliverxu@tencent.com

Thomas Graf
Swisscom
Binzring 17
CH-8045 Zürich
Switzerland
Email: thomas.graf@swisscom.com

Yunan Gu
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: guyunan@huawei.com

Shunwan Zhuang
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: zhuangshunwan@huawei.com

Zhenbin Li
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: lizhenbin@huawei.com