Network Working Group
Internet-Draft
Intended status: Best Current Practice
Expires: August 12, 2021

M. McBride
Futurewei
D. Madory
Kentik
J. Tantsura
Apstra
R. Raszuk
Bloomberg LP
H. Li
HPE
J. Heitz
Cisco

February 8, 2021

AS Path Prepending
draft-ietf-grow-as-path-prepending-03

Abstract

AS Path Prepending provides a tool to manipulate the BGP AS_Path attribute through prepending multiple entries of an AS. AS Path Prepending is used to deprioritize a route or alternate path. By prepending the local ASN multiple times, ASs can make advertised AS paths appear artificially longer. Excessive AS Path Prepending has caused routing issues in the internet. This document provides guidance,to the internet community, with how best to utilize AS Path Prepending in order to avoid negatively affecting the internet.

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 August 12, 2021.

Table of Contents

1. Introduction

The Border Gateway Protocol (BGP) [RFC4271] specifies the AS_PATH attribute which enumerates ASs a route update has traversed. If the UPDATE message is propagated over an external link, then the local AS number is prepended to the AS_PATH attribute, and the NEXT_HOP attribute is updated with an IP address of the router that should be used as a next hop to the network. If the UPDATE message is propagated over an internal link, then the AS_PATH attribute and the NEXT_HOP attribute are passed unmodified.

A common practice among operators is to prepend multiple entries of an AS (known as AS Path Prepending) in order to deprioritize a route or a path. This has worked well in practice but the practice is increasing, with both IPv4 and IPv6, and there are inherit risks to the global internet especially with excessive AS Path Prepending. Prepending is frequently employed in an excessive manner such that it renders routes vulnerable to disruption or misdirection. AS Path Prepending is discussed in Use of BGP Large Communities [RFC8195] and this document provides additional, and specific, guidance to operators on how to be a good internet citizen with the proper use of AS Path Prepending.

1.1. Requirements Language

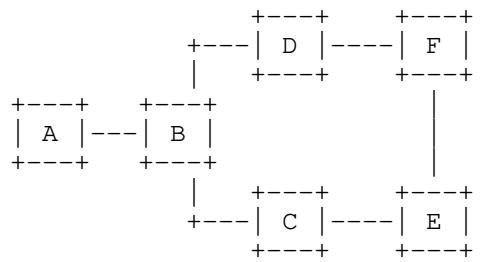
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Use Cases

There are various reasons that AS Path Prepending is in use today including:

- Preferring one ISP over another ISP on the same ASBR or across different ASBRs
- Preferring one ASBR over another ASBR in the same site
- Utilize one path exclusively and another path solely as a backup
- Signal to indicate that one path may have a different amount of capacity than another where the lower capacity link still takes traffic
- An ISP doesn’t accept traffic engineering using BGP communities. Prepending is the only option.

The following illustration, from Geoff Hustons Path Prepending in BGP [1], shows how AS Prepending is typically used:



B will normally prefer the path via C to send traffic to E, as this represents the shorter AS path for B. If E were to prepend a further two instances of its own AS number when advertising its routes to C, then B will now see a different situation, where the AS Path via D represents the shorter path. Through the use of selective prepending E is able to alter the routing decision of B, even though B is not an adjacent neighbour of E. The result is that traffic from A and B will be passed via D and F to reach E, rather than via C. In this way prepending implements action at a distance where the routing decisions made by non-adjacent ASs can be influenced by selective AS Path prepending.

3. Problems

Since it is so commonly used, what is the problem with the excessive use of AS Path Prepending? Here are a few examples:

3.1. Excessive Prepending

The risk of excessive use of AS Path Prepending can be illustrated with real-world examples that have been anonymized using documention prefixes [RFC5737] and ASs [RFC5398] . Consider the prefix 198.51.100.0/24 which is normally announced with an inordinate amount of prepending. A recent analysis revealed that 198.51.100.0/24 is announced to the world along the following AS path:

64496 64511

In this example, the origin AS64511 appears 23 consecutive times before being passed on to a single upstream (AS64496), which passes it on to the global internet, prepended-to-all. An attacker, wanting to intercept or manipulate traffic to this prefix, could enlist a datacenter to allow announcements of the same prefix with a fabricated AS path such as 999999 64496 64511. Here the fictional

AS999999 represents the shady datacenter. This malicious route would be preferred due to the shortened AS path length and might go unnoticed by the true origin, even if route-monitoring had been implemented. Standard BGP route monitoring checks a route's origin and upstream and both would be intact in this scenario. The length of the prepending gives the attacker room to craft an AS path that would appear plausible to the casual observer, comply with origin validation mechanisms, and not be detected by off-the-shelf route monitoring.

3.2. Prepending during a routing leak

In April 2010, a service provider experienced a routing leak. While analyzing the leak something peculiar was noticed. When we ranked the approximately 50,000 prefixes involved in the leak based on how many ASs accepted the leaked routes, most of the impact was constrained to Country A routes. However, two of the top five most-propagated leaked routes (listed in the table below) were Country B routes.

During the routing leak, nearly all of the ASs of the internet preferred the Country A leaked routes for 192.0.2.0/21 and 198.51.100.0/22 because, at the time, these two Country B prefixes were being announced to the entire internet along the following excessively prepended AS path: 64496 64500 64511 64511 64511 64511 64511 64511. Virtually any illegitimate route would be preferred over the legitimate route. In this case, the victim is all but ensuring their victimhood.

There was only a single upstream seen in the prepending example from above, so the prepending was achieving nothing except incurring risk. You would think such mistakes would be relatively rare, especially now, 10 years later. As it turns out, there is quite a lot of prepending-to-all going on right now and during leaks, it doesn't go well for those who make this mistake. While one can debate the merits of prepending to a subset of multiple transit providers, it is difficult to see the utility in prepending to every provider. In this configuration, the prepending is no longer shaping route propagation. It is simply incentivizing ASs to choose another origin if one were to suddenly appear whether by mistake or otherwise.

3.3. Prepending to All

Based on analysis done in 2019, Excessive AS Path Prepending [2], out of approximately 750,000 routes in the IPv4 global routing table, nearly 60,000 BGP routes are prepended to 95% or more of hundreds of BGP sources. About 8% of the global routing table, or 1 out of every 12 BGP routes, is configured with prepends to virtually the entire

internet. The 60,000 routes include entities of every stripe: governments, financial institutions, even important parts of internet infrastructure.

Much of the worst propagation of leaked routes during big leak events have been due to routes being prepended-to-all. AS64505 leak of April 2014 (>320,000 prefixes) was prepended-to-all. And the AS64506 leak of June 2015 (>260,000 prefixes) was also prepended-to-all. Prepended-to-all prefixes are those seen as prepended by all (or nearly all) of the ASs of the internet. In this configuration, prepending is no longer shaping route propagation but is simply incentivizing ASs to choose another origin if one were to suddenly appear whether by mistake or otherwise. The percentage of the IPv4 table that is prepended-to-all is growing at 0.5% per year. The IPv6 table is growing slower at 0.2% per year. The reasons for using prepend-to-all appears to be due to 1) the AS forgetting to remove the prepending for one of its transit providers when it is no longer needed and 2) the AS attempting to de-prioritize traffic from transit providers over settlement-free peers and 3) there are simply a lot of errors in BGP routing. Consider the prepended AS path below:

64496 64501 64501 64510 64510 64501 64510 64501 64501 64510 64510 64501 64501 64510

The prepending here involves a mix of two distinct ASNs (64501 and 64510) with the last two digits transposed.

3.4. Memory

Long AS Paths cause an increase in memory usage by BGP speakers. The memory usage is not so much a concern in the control plane BGP implementations, but more so when AS Paths are included in Netflow messages. Netflow is processed in the forwarding plane, where memory is more expensive than in the control plane.

A concern about an AS Path longer than 255 is the extra complexity required to process it, because it needs to be encoded in more than one AS_SEQUENCE in the AS_PATH BGP path attribute.

3.5. Errant announcement

There was an Internet-wide outage caused by a single errant routing announcement. In this incident, AS64496 announced its one prefix with an extremely long AS path. Someone entered their ASN instead of the prepend count 64496 modulo 256 = 252 prepends and when a path lengths exceeded 255, routers crashed

4. Alternatives to AS Path Prepend

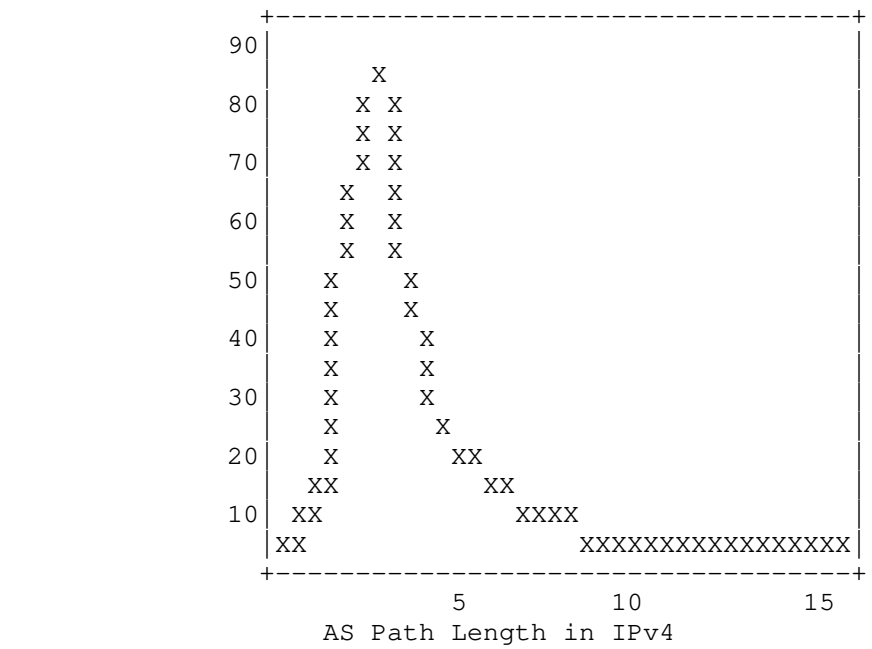
There are various options to provide path preference without needing to use AS Path Prepend:

- Use predefined communities that are mapped to a particular behavior when propagated.
- Announce more specific routes on the preferred path.
- The BGP Origin Code is an attribute that is used for path selection and can be used as a high order tie-breaker. The three origin codes are IGP, EGP and INCOMPLETE. When AS Paths are of equivalent length, users could advertise paths, with IGP or EGP origin, over the preferred path while the other ASBRs (which would otherwise need to prepend N times) advertises with an INCOMPLETE origin code.

5. Best Practices

Many of the best practices, or lack thereof, can be illustrated from the preceeding examples. Here's a summary of the best current practices when using AS Path Prepending:

- Network operators should ensure prepending is absolutely necessary as many networks have excessive prepending. It is best to innumerate what the routing policies are intended to achieve before concluding that prepending is a solution
- The neighbor you are prepending may have an unconditional preference for customer routes and prepending doesn't work. It's helpful to check with neighbors to see if they will honor the prepend to avoid wasting effort and potentially causing further vulnerabilities.
- There is no need to prepend more than 5 ASs. The following diagram shows that, according to Excessive AS Path Prepending [3], 90% of AS path lengths are 5 ASNs or fewer in length.



X Axis = unique AS Paths in millions

- Don’t prepend ASNs that you don’t own.

- Prepending-to-all is a self-inflicted and needless risk that serves little purpose. Those excessively prepending their routes should consider this risk and adjust their routing configuration.

- The Internet is typically around 5 ASs deep with the largest AS_PATH being 16-20 ASNs. Some have added 100 or more AS Path Prepends and operators should therefore consider limiting the maximum AS-path length being accepted through aggressive filter policies.

6. IANA Considerations

7. Security Considerations

Long prepending may make a network more vulernable to route hijacking which will exist whenever there is a well connected peer that is willing to forge their AS_PATH or allows announcements with a fabricated AS path.

8. Acknowledgement

The authors would like to thank Greg Skinner, Randy Bush, Dave Farmer, Nick Hilliard, Martijn Schmidt, Michael Still, Geoff Huston and Jeffrey Haas for contributing to this document.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <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>.

[RFC5398] Huston, G., "Autonomous System (AS) Number Reservation for Documentation Use", RFC 5398, DOI 10.17487/RFC5398, December 2008, <https://www.rfc-editor.org/info/rfc5398>.

[RFC5737] Arkko, J., Cotton, M., and L. Vegoda, "IPv4 Address Blocks Reserved for Documentation", RFC 5737, DOI 10.17487/RFC5737, January 2010, <https://www.rfc-editor.org/info/rfc5737>.

[RFC8195] Snijders, J., Heasley, J., and M. Schmidt, "Use of BGP Large Communities", RFC 8195, DOI 10.17487/RFC8195, June 2017, <https://www.rfc-editor.org/info/rfc8195>.

9.2. URIs

[1] https://labs.apnic.net/?p=1264

[2] https://blogs.oracle.com/internetintelligence/excessive-as-path-prepending-is-a-self-inflicted-vulnerability

[3] https://blogs.oracle.com/internetintelligence/excessive-as-path-prepending-is-a-self-inflicted-vulnerability

Authors' Addresses

Mike McBride
Futurewei

Email: michael.mcbride@futurewei.com

Doug Madory
Kentik

Email: dmadory@kentik.com

Jeff Tantsura
Apstra

Email: jefftant.ietf@gmail.com

Robert Raszuk
Bloomberg LP

Email: robert@raszuk.net

Hongwei Li
HPE

Email: flycoolman@gmail.com

Jakob Heitz
Cisco
170 West Tasman Drive
San Jose, CA 95134
USA

Email: jheitz@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 26, 2021

T. Graf
Swisscom
P. Lucente
NTT
P. Francois
INSA-Lyon
Y. Gu
Huawei

February 22, 2021

BMP (BGP Monitoring Protocol) Seamless Session
draft-tppy-bmp-seamless-session-00

Abstract

This document describes an optional BMP session lifecycle extension to prevent data duplication of previously exported messages when TCP session is re-established. It prevents loss of messages between TCP session re-establishments and increase overall BMP scalability.

Table of Contents

1. Introduction

With the constant increase of BGP paths, the increase of BMP BGP RIB coverage from RFC8671 [RFC8671] and draft-ietf-grow-bmp-local-rib [I-D.ietf-grow-bmp-local-rib], the addition of new TLVs such as draft-cppy-grow-bmp-path-marking-tlv [I-D.cppy-grow-bmp-path-marking-tlv] and draft-xu-grow-bmp-route-policy-attr-trace [I-D.xu-grow-bmp-route-policy-attr-trace], more BMP messages and BGP contexts, such as peering, route-policy or RIB, are exported from BMP client to server.

With each BMP session re-establishment, clients export the initial BGP RIB via BMP route-monitoring messages as described in section 5 of RFC7854 [RFC7854]. Regardless if the same messages were already exported in a previous BMP session or not. This leads to data duplication and unnecessary strain of the BMP client and server.

In a network most times BMP sessions are re-established within a short period of time due to connectivity interruption between BMP client and server or restart of the BMP server due to maintenance. Even though most BMP client implementations support a BMP buffering mechanism, messages are not buffered across BMP session re-establishment, thus leading to a loss of messages.

Therefore, the proposed BMP session lifecycle improvement covers

- Brief loss of connectivity between BMP client and server
- Seamless Maintenance of BMP server

It is based on RFC7413, TCP Fast Open [RFC7413], which allows previously established TCP transport sessions to be re-established more efficiently.

This draft describes how the BMP application MUST behave during TCP transport re-establishment period in order to prevent metric loss.

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

3. Definitions

Brief loss of connectivity between BMP client and server: Describes a period of time, in seconds, starting from the point in time in which the BMP client detects loss of connectivity to the BMP server and tries to re-establish the TCP session.

Maintenance of BMP server: Describes a period of time, in seconds, starting from when the BMP server daemon is restarted for maintenance purposes and the BMP client tries to re-establish the TCP session.

4. BMP Client and Server Capability

To support brief loss of connectivity between BMP client and server, the BMP client and server MUST support TCP Fast Open as described in RFC7413 [RFC7413].

To support seamless maintenance of a BMP server, the BMP client and server MUST support TCP Fast Open as described in RFC7413 [RFC7413] and the restart of the BMP server MUST distinguish between normal and seamless restart, wherever TCP Fast Open cookies are preserved or not.

5. Updated BMP Session Lifecycle

Section 3 of RFC7413 [RFC7413] describes the TCP Fast Open extension in the initial TCP SYN packet and the cookie handling during initial and subsequent re-establishment of the TCP transport session.

Section 3.3 of RFC7854 [RFC7854] describes that the BMP session closes with the TCP session. This behavior is extended with a configurable BMP session timeout.

The BMP session timeout starts counting down under the following conditions:

- Configured value is bigger than 0
- Current TCP session was established with Fast Open extension and cookie has been saved
- BMP buffer is not full
- TCP session is going to be terminated

The default BMP session timeout is 60 seconds.

While the time is counting down, all the BMP messages, regardless of message type, MUST be buffered. At this stage, the BMP session is still considered to be alive.

When a TCP session is re-established with TCP Fast Open extension and the cookie is identical to the previous TCP session with the same BMP peer, the BMP session remains alive, BMP buffer is exported and normal operation continues.

When a TCP session is re-established without TCP Fast Open extension or with TCP Fast Open extension but the cookie is not identical to the previous TCP session with the same BMP peer, the BMP session is considered terminated and starts with a new BMP Initiation message.

When a TCP session is not re-established within the configured timeout, then the BMP buffer is discarded and the BMP session is considered terminated.

When the BMP buffer is full before the TCP session is re-established, then the BMP buffer is discarded and the BMP session is considered terminated.

6. Security Considerations

The same security considerations apply as for TCP Fast Open RFC7413 [RFC7413].

7. Operational Considerations

From the perspective of the BMP server, the TCP Fast Open mechanism is rather transparent since it is entirely handled by the operating system kernel: this also means a BMP Server application can't determine if the TCP session was established with SYN Cookies or without them.

Upon terminating the existing BMP session(s), the BMP server should dump to persistent storage the BGP RIBs currently in memory. In terms of encoding, MRT format could be used for the task (ie. draft-petrie-grow-mrt-bmp)

At restart, the BMP server should first restore the content of BGP RIBs from persistent storage before accepting any incoming connection from BMP clients. Only once this process is finished, connections can then be accepted again so that messages buffered by BMP clients are applied to the last known BGP RIBs upon termination.

8. References

8.1. Normative References

[RFC7413] Cheng, Y., Chu, J., Radhakrishnan, S., and A. Jain, "TCP Fast Open", RFC 7413, DOI 10.17487/RFC7413, December 2014, <https://www.rfc-editor.org/info/rfc7413>.

[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>.

8.2. Informative References

[I-D.cppy-grow-bmp-path-marking-tlv] Cardona, C., Lucente, P., Francois, P., Gu, Y., and T. Graf, "BMP Extension for Path Status TLV", draft-cppy-grow-bmp-path-marking-tlv-07 (work in progress), October 2020.

[I-D.ietf-grow-bmp-local-rib] Evens, T., Bayraktar, S., Bhardwaj, M., and P. Lucente, "Support for Local RIB in BGP Monitoring Protocol (BMP)", draft-ietf-grow-bmp-local-rib-09 (work in progress), January 2021.

[I-D.xu-grow-bmp-route-policy-attr-trace] Xu, F., Graf, T., Gu, Y., Zhuang, S., and Z. Li, "BGP Route Policy and Attribute Trace Using BMP", draft-xu-grow-bmp-route-policy-attr-trace-05 (work in progress), July 2020.

[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

Thomas Graf
Swisscom
Binzring 17
Zurich 8045
Switzerland

Email: thomas.graf@swisscom.com

Paolo Lucente
NTT
Siriusdreef 70-72
Hoofddorp, WT 2132
Netherlands

Email: paolo@ntt.net

Pierre Francois
INSA-Lyon
Lyon
France

Email: Pierre.Francois@insa-lyon.fr

Yunan Gu
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: guyunan@huawei.com