As the number of operational use cases of BGP grows there is demand to increase the level of separation and processing independence between various address families carried by BGP today. This document augments base BGP specification in allowing local configuration of BGP port number by the operator to run parallel fully disjoined BGP instances allowing full processing separation between them.

While local BGP implementation may already assure BGP process or thread robustnes the general aim here is to allow similar level of groups of BGP address families independence when running BGP code on general purpose hardware as well as x86 based route reflectors.
Internet-Draft mi-bgp October 2018

The current document has been created based on parent document Transport Instance BGP. Below is the list of contributors who contributed to the previous document:

Keyur Patel, Arrcus Inc., 2077 Gateway Place Suite 400, San Jose, CA, 95119, US, Email: keyur@arccus.com

Bruno Decraene, France Telecom, 38 rue du General Leclerc, Issy Moulineaux cedex 9, France, Email: bruno.decraene@orange-ftgroup.com

Jakob Heitz, Cisco, 170 Tasman Dr, San Jose, CA 95134, US Email: jheitz@cisco.com

Thomas D. Nadeau

Jie Dong, Huawei Technologies Co.,Ltd, KuiKe Building, No.9 Xinxi Rd., Beijing, Hai-Dian District, 100085, P.R. China, Email: dongjie_dj@huawei.com
2. Introduction

BGP4 [RFC4271] protocol is practically a single standard today for the distribution of inter-domain routing information. Under many applications, it is also used as the protocol of choice when disseminating various application-based information intra-domain. Its popularity and wide use has been effectively provided by its extensibility, reliable transport, session protection as well as built-in loop prevention mechanisms.

It has been observed in both intra-domain as well as inter-domain applications that reliable information distribution is an extremely desired tool for many applications. The introduction of Multiprotocol extensions to BGP [RFC4760] made it appealing for new kinds of information to be carried over BGP4.

While these extensions have proven to be useful, they however have increased the load of information as well as the type of information that BGP was originally envisioned to carry. As example carrying link state LSDB and TED databases with BGP-LS [RFC7752] with all its extensions.

This draft proposes to group together similar classes of information carried in BGP and execute it by separate process on a per AFI/SAFI basis. Each such BGP process would be fully independent from each other and will listen on the TCP port as chosen by local configuration by the operator.

This draft proposes that the current BGP infrastructure will continue to be used to disseminate Internet routing related information and will continue to listen on BGP TCP port 179. Non Internet routing information or private routing data are recommended to be handled by parallel BGP processes listening on their own TCP ports.

Another point of view in favor of BGP instance separation is the aspect of service protection. One could see BGP process responsible for global routing due to its global nature, much more exposed to control plane errors and attacks than potentially private only BGP instance contained to one of few ASes, possibly under common administration. In the same way one could also observe that by fully separating global Internet BGP from any local BGP based services the Internet itself can be fully isolated from any issues caused by local service provider’s services.
3. Today’s operation

In today’s networks BGP4 operates per BGP specification [RFC4271]. This model of operation has proven to have number of disadvantages when it comes to concurrent support of multiple applications when amount of transported number of entries is already non trivial, when is not bounded by application architecture and when it is continuously growing.

There are many examples where major router vendors recommend to separate route reflectors into disjoined clusters so Internet routes are not affected by L3VPN routes or BGP-LS data. To put things into right perspective one needs to observe that local per box scaling numbers have already reached millions of VPN routes. BGP-LS is carrying more and more IGP and TE data almost every day.

4. Related work

To address the session separation without forcing users to manually bound each session or group of session to a different BGP peering address Multisession BGP [I-D.ietf-idr-bgp-multisession] solution has been proposed. It is our opinion that Multisession BGP is a tool to automatically bound group of applications to different TCP sessions however code complexity of handling this automation has resulted in significant complications and in number of BGP code bases has been disabled or even removed.

Observation also needs to be made that in this model all BGP OPEN messages would still end up going to the same BGP TCP port number 179. Furthermore, all the incoming sessions would be handled by the same BGP process. Number of applications BGP carries data for today call for much stronger separation including not just session but processing isolation and operator’s control.

Multisession if still required could be applicable within each major BGP instance by providing fine granularity of session separation within a given instance.

5. Multi Instance Proposal

In order to minimize impact between different classes of applications carried today or to be carried by BGP in the future to those of critical nature for Internet connectivity, this document proposes to run separate instances of BGP for each of them.

The separation of concurrent, but not necessarily congruent BGP instances will be complete. Each such instance can run the same or different bgp code base.
5.1. Protocol changes

The proposed here Multi Instance BGP does not require any changes to BGP4 protocol mechanism, state machine, error handling or operation. The exact same procedures and semantics apply in the same way for routing instance as well as for other instances of BGP. The operational advantage in the instance separation is the ability to apply different process level parameters and tuning for each launched instance precisely fitting to the operator’s needs.

The only protocol change proposed in this document is the ability to specify locally TCP port number given BGP instance will be listening on.

5.2. BGP Identifier BGP peering address

When running both independent instances on the same platform question arises on the recommended choice for BGP Identifier [RFC6286] as well as BGP peering address to be used.

It needs to be observed that since via different BGP OPEN TCP port number and then different session ports if only implementation allows there is no requirement this specification would enforce to make any of those different between any instances.

Another advantage of sharing the same peering address of BGP sessions between instances is that in the event of operator’s choice to use fast failure detection tools like BFD [RFC5880] the same event can be passed to both instances without any additional need to run two parallel and independent BFD sessions.

6. Summary of benefits

Below is a combined list of main benefits provided by Multi Instance BGP:

- Isolation and independence from protocol or process failures caused by any instance.
- Independence in: CPU usage, memory space and internal platform’s resources.
- Different port for BGP OPEN messages allowing the same BGP router_id or peering address sharing between instances.
- Different and fully isolated TCP sessions between instances. Each instance may still benefit from multisessions BGP proposal within each instance.
Possibility of different IP precedence BGP message marking for more fair and accurate PHB treatment.

7. Applications

Recent BGP extensions RFC7752 [RFC7752] define a way to carry link state information over BGP transport. Fate sharing of such new type of data with existing BGP address families is mutually risky. While originally intended to be physically separated operational experience proves that very often it is used just as an "add-on" to existing data already carried over BGP including being handled by the same set of BGP infrastructure (ex: route reflectors). Separating BGP-LS to travel over disjoined TCP sessions will not only reduce mutual impact, but also enable separate processing of it (ex: pinning to different CPU core) even on the same BGP hardware.

Another group of potential candidates for Multi Instance BGP could be any type of auto discovery mechanism for other applications or for BGP itself [I-D.raszuk-idr-bgp-auto-discovery]. Other examples may include: L2VPN/VPLS or MVPN Auto Discovery [RFC6513] as possible candidates.

Another class of applications perfectly fitting the separate BGP instance model for it’s global information distribution authors foresee a mapping plane of identifiers to locators in the new evolving internet architecture. As example LISP-ALT [RFC6836] is already calling to use BGP as a mapping plane protocol to simplify initial deployment. While it is foreseen that in the future those may migrate to better distribution schemes for example LISP-DHT to get enough of initial traction and momentum a Multi Instance BGP seems like a very good match to the mapping plane requirements.

One may observe that Service Providers may choose to deploy a new instances of BGP to carry their critical services (example L3VPNs) over it for full isolation from Internet BGP. In such application they will be able to prioritize such instance according to their internal policy and offered services prioritization.

8. Security considerations

Multi Instance BGP proposed in this document does not introduce any new security concerns as compared to base BGP4 specification [RFC4271]. Also all security work applicable to base routing instance BGP does also apply as is to transport instance BGP.
9. IANA Considerations

This document presents no requirements to IANA.

10. Acknowledgments

The authors would like to thank Randy Bush, Tom Scholl and Joel Halpern for their valuable comments provided to the parent document - BGP Transport Instance.

11. References

11.1. Normative References


11.2. Informative References


[I-Dietf-ospf-transport-instance]

[I-D.raszuk-idr-bgp-auto-discovery]


Author’s Address

Robert Raszuk (editor)
Bloomberg LP
731 Lexington Ave
New York, NY 10022
US

Email: robert@raszuk.net