

# **Transport Instance BGP version 00**

*draft-raszuk-ti-bgp-00*

*Robert Raszuk, Keyur Patel, Bruno Decraene,  
Jakob Heitz, Thomas Nadeau, Jie Dong  
& Yoshinobu Matsuzaki*

# Agenda:

- **Background**
- **Proposal**
- **Discussion**

# Background ....

- **BGP4 provides a very good platform to carry inter and intra-domain various sorts of information. Today we have:**

SAFI	Application	Reference
1	NLRI used for unicast forwarding	RFC4760
2	NLRI used for multicast forwarding	RFC4760
4	NLRI with MPLS Labels	RFC3107
5	MCAST-VPN	draft-ietf-l3vpn-2547bis-mcast-bgp
6	NLRI used for Dynamic Placement of Multi-Segment	draft-ietf-pwe3-dynamic-ms-pw
7	Encapsulation SAFI	RFC5512
64	Tunnel SAFI	Nalawade
65	Virtual Private LAN Service (VPLS)	RFC4761
66	BGP MDT SAFI	Nalawade
67	BGP 4over6 SAFI	Cui
68	BGP 6over4 SAFI	Cui
69	Layer-1 VPN auto-discovery information	RFC-ietf-l1vpn-bgp-auto-discovery-05.txt
128	MPLS-labeled VPN address	RFC2547
129	Multicast for BGP/MPLS IP Virtual Private Networks (VPNs)	
132	Route Target constrains	RFC4684
133	IPv4 dissemination of flow specification rules	RFC-ietf-idr-flow-spec-09.txt
134	VPNv4 dissemination of flow specification rules	RFC-ietf-idr-flow-spec-09.txt
140	VPN auto-discovery	draft-ietf-l3vpn-bgpvpn-auto
241-254	Private Use	RFC4760

# Background ....

- Out of this list only a few SAFIs are related to Internet routing:

SAFI	Application	Reference
1	NLRI used for unicast forwarding	RFC4760
2	NLRI used for multicast forwarding	RFC4760
4	NLRI with MPLS Labels	RFC3107
5	MCAST-VPN	draft-ietf-l3vpn-2547bis-mcast-bgp
6	NLRI used for Dynamic Placement of Multi-Segment	draft-ietf-pwe3-dynamic-ms-pw
7	Encapsulation SAFI	RFC5512
64	Tunnel SAFI	Nalawade
65	Virtual Private LAN Service (VPLS)	RFC4761
66	BGP MDT SAFI	Nalawade
67	BGP 4over6 SAFI	Cui
68	BGP 6over4 SAFI	Cui
69	Layer-1 VPN auto-discovery information	RFC-ietf-l1vpn-bgp-auto-discovery-05.txt
128	MPLS-labeled VPN address	RFC2547
129	Multicast for BGP/MPLS IP Virtual Private Networks (VPNs)	
132	Route Target constrains	RFC4684
133	IPv4 dissemination of flow specification rules	RFC-ietf-idr-flow-spec-09.txt
134	VPNv4 dissemination of flow specification rules	RFC-ietf-idr-flow-spec-09.txt
140	VPN auto-discovery	draft-ietf-l3vpn-bgpvpn-auto
241-254	Private Use	RFC4760

- Another point to make is that different applications may have different requirements (in term of scalability, response time, importance for SP / end customers...)

# Background ....

- **This proposal is not about „BGP overload” → Authors do not believe BGP is overloaded**
- **This proposal is about separation of various types of information being carried by BGP today and tomorrow from each other.**
- **The issue that such separation is required have been stated in various applications which do recommend physical hardware or instance separation. Example:**

```
Source: draft-ietf-l3vpn-mvpn-considerations
```

```
In the higher scale scenarios, it may be required to adapt the route reflector infrastructure to the mVPN routing load by using, for example:
```

- o a separation of resources for unicast and multicast VPN routing :  
using dedicated mVPN Route Reflector(s) (or using dedicated mVPN BGP sessions or dedicated mVPN BGP instances);

# Background ....

- **Proposed separation can be used in multiple ways:**
  - **To run independently on the same BGP platforms Internet routing and non Internet routing. Achieves protection from Internet to commercial services (example: DDoS), as well as protection of Internet itself from any local service related churn or impact.**
  - **To run independently on the same BGP platforms opaque applications which are only carried by BGP for convenience (example: auto-discovery, namespace separation, etc)**
  - **To achieve session and instance full protection from each other's failures or network attacks.**

# Agenda:

- **Background**
- **Proposal**
- **Discussion**

# Proposal ....

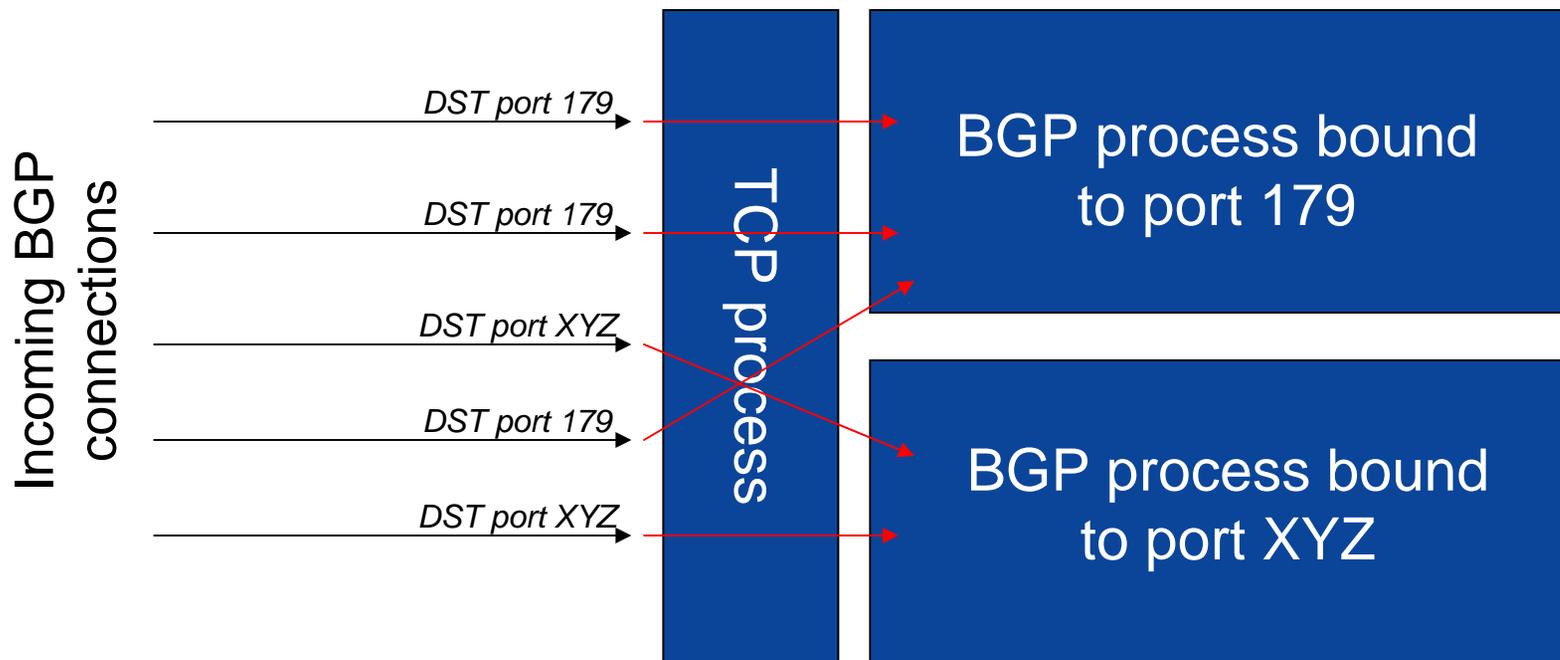
- **This document proposes the creation of second BGP instance to allow for clear separation between BGP based applications on a per operator's choice**
- **No impact to IGP ... No need to inject new peering addresses ... by defining new TCP/SCTP port numbers to be used by the second instance**
- **Ability to run the same application in both instances for easy migrations**
- **No change to BGP protocol other than port number for initial session establishment**
- **No new security concerns**

## Proposal ....

- **The level of flexibility to tune each of those instances depends on the implementation choice. Examples:**
  - **Max CPU processing time**
  - **Max memory capacity**
  - **Different tunable sizing of BGP I/O queues**
  - **Manual locking to preferred CPU core**
  - **Per instance choice of IP precedence in messages**
  - **Per instance tunable TCP parameters**
- **By de-multiplexing at different initial OPEN port the same BGP peering addresses as well as BGP Identifier can be shared by both instances. The choice is left for the operator's local choice.**

# Proposal ....

- The clear and straightforward way for bounding incoming TCP connections to corresponding BGP deamons



# Agenda:

- **Background**
- **Proposal**
- **Discussion**

# Discussion ....

- Why only two instances ?

→ Authors believe that at present time there is sufficient case made for two instances. If someone can bring solid arguments for more instances discussion is welcome.

- How is this proposal compared with multiseession -04 proposal ?

→ draft-ietf-idr-bgp-multiseession defines set of procedures for separation of actual sessions within single connection. Version -04 adds ability to initially OPEN a session on port 179 then close it with the hint of establishing a new one on a new dynamically assigned arbitrary TCP port.

This document defines separation of BGP instances such sessions would directly terminate at without any dependency between such instances. Both documents could be complimentary (perhaps subject to be merged) and it is expected that multiseession could be used within each BGP instance when required.

Running BGP on dynamic and unknown port numbers seems like quite difficult troubleshooting wise. Non of the deployed sniffers would be able to capture BGP protocol data streams.

## Discussion ....

- **Would such separation be possible by an implementation ?**

**→ An Implementation can offer a lot of separation between particular SAFIs. This proposals brings two main additional values:**

- 1. Enables operators and not vendors to make an actual choice of separation driven by their own set of business and technical priorities**
- 2. Defines a new port numbers for true full separation and easy deployment**

**Thank you.**