Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: September 29, 2011

J. Scudder
Juniper Networks
C. Appanna
Cisco Systems
I. Varlashkin
Easynet Global Services

March 28, 2011

Multisession BGP
draft-ietf-idr-bgp-multisession-06

Abstract

This specification augments "Multiprotocol Extensions for BGP-4" (MP-BGP) by proposing a mechanism to facilitate the use of multiple sessions between a given pair of BGP speakers. Each session is used to transport routes related by some session-based attribute such as AFI/SAFI. This provides an alternative to the MP-BGP approach of multiplexing all routes onto a single connection.

Use of this approach is expected to provide finer-grained fault management and isolation as the BGP protocol is used to support more and more diverse services.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Most BGP [RFC4271] implementations only permit a single ESTABLISHED connection to exist with each peer. More precisely, they only permit a single ESTABLISHED connection for any given pair of IP endpoints.

BGP Capabilities [RFC5492] extend BGP to allow diverse information (encoded as "capabilities") to be associated with a session. In some cases, a capability may relate to the operation of the protocol machinery; an example is Route Refresh [RFC2918]. However, in other cases a capability may relate specifically to some common distinguishing characteristic of the routes carried over the session; an example is Multiprotocol BGP [RFC4760].

Multiprotocol BGP [RFC4760] extends BGP to allow information for multiple NLRI families and sub-families to be transported in BGP. Routes for different families are distinguished by AFI and SAFI. Routes for different families are commonly multiplexed onto a single BGP session.

A common criticism of BGP is the fact that most malformed messages cause the session to be terminated. While this behavior is necessary for protocol correctness, one may observe that the protocol machinery of a given implementation may only be defective with respect to a given AFI/SAFI. Thus, it would be desirable to allow the session related to that family to be terminated while leaving other AFI/SAFI unaffected. As BGP is commonly deployed, this is not possible.

A second criticism of BGP is that it is difficult or in some cases impossible to manage control plane resource contention when BGP is used to support diverse services over a single session. In contrast, if a single BGP session carries only information for a single service (or related set of services) it may be easier to manage such contention.

In this specification, we propose a mechanism by which multiple transport sessions may be established between a pair of peers. Each transport session is identified by a distinct set of BGP capabilities, notably the MP-BGP capability.

Each session is distinct from a BGP protocol point of view; an error or other event on one session has no implications for any other session. All protocol modifications proposed by this specification take place during the OPEN exchange phase of the session, there are no modifications to the operation of the protocol once a session reaches ESTABLISHED state.

Although AFI/SAFI is perhaps the most obvious way to group sets of

routes being exchanged between BGP peers, sessions can also be distinguished by other BGP capabilities. In general, any capability used in this fashion would be expected to have semantics of identifying some common distinguishing characteristic of a set of routes, just as AFI/SAFI does; however, specifics are beyond the scope of this document. Most examples in this document are focusing on MP-BGP capability (or interchangeably, AFI/SAFI) based grouping for simplicity reason. However actual application of multisessions extension . Such use is illustrative and is not intended to be limiting.

Routers implementing this specification MUST also implement the base criteria that is used to define sessions. For example if AFI/SAFI based sessions are desired then routers implementing this specification MUST also implement MP-BGP [RFC4760].

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. Definitions

"MP-BGP capability" refers to the capability [RFC5492] with code 1, specified in MP-BGP [RFC4760] section 8.

A BGP speaker is said to "support" some feature or functionality (for example, to support this specification, or to support a particular AFI/SAFI) when the BGP implementation supports the feature AND the feature has not been disabled by configuration.

The Session Identifier is a capability or group of capabilities that will be used to differentiate individual BGP sessions between two IP endpoints. When the AFI/SAFI is used to distinguish sessions, the MP-BGP capability is the session identifier.

3. Overview of operations

To allow multiple sessions between same pair of BGP speakers to co-exist BGP Multisession extension modifies Connection Collision Detection procedure of the base BGP specification (RFC4271). Rather than considering only IP addresses of the peers new procedure also takes into account list of certain session attributes, such as AFI/SAFI, to determine uniqueness of the sessions. When sessions are deemed to be unique each of them is then handled independently,

therefore critical conditions (such as malformed UPDATEs) in one session won't affect others.

BGP Multisession extension introduces new BGP capability code to indicate that a BGP speaker supports protocol modification described in this document and new error message sub-codes that facilitate handling of incompatible configurations between two speakers.

Following sections provide formal description of the protocol enhancement. Additionally, Appendix contains non-normative examples of desired behaviour for Multisession-enabled BGP speakers, which is intended only for illustrative purpose.

4. Multisession BGP Capability Code

This specification defines the Multisession capability [RFC5492]:

Capability code (1 octet): 68

Capability length (1 octet): variable

Capability value (1 octet): Flags followed by the list of capabilities that define a session.

```
          0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
         |G|  Reserved    |  Session Id   ~
         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

G - the most significant bit was originally intended by earlier draft version of Multisession specification to denote capability of a BGP speaker to group multiple capability values into one session. As this information can be deduced from Session Id, the use of G bit is deprecated - implementations conforming to final version of Multisession specification SHOULD NOT rely on value of the G bit.

Reserved - MUST be set to zero by sender, MUST be ignored by receiver

Session Id(entifier) - list of zero or more capability codes (1 octet each) defined in BGP, whose values will be used to distinguish one group from another. The size of the list is inferred from the length of the overall capability; it is the capability length minus one. The Multisession capability code itself MUST NOT be listed; if listed it MUST be ignored upon receipt.

Empty Session Id list and Session Id containing 1 (one, Multiprotocol

Extensions) as the only value are considered equal and indicate that AFI/SAFI list in the OPEN message is used to distinguish the groups. However, if BGP speaker wishes to use compound Session Id that includes AFI/SAFI list as one of the components, then Capability Code 1 MUST be explicitly included in the Session Id. For example, if BGP speaker Session Id to 'X' (denoting Capability Foo) then only Foo will be used as Session Id, i.e. session where Foo is 1 and AFI/SAFI is 1/1 and session where Foo is 1 and AFI/SAFI is 1/2 will be considered as conflicting. On the other hand Session Id set to '1 X' or 'X 1' indicates that groups are identified by combination of Foo and AFI/SAFI, i.e. above two sessions as well as session where Foo is 2 and AFI/SAFI is 2/4 will be considered unique.

For given pair of BGP peers Multisession capability MUST be used either on all or none sessions. This is required due to different connection collision handling procedure used by multisession.

5. New NOTIFICATION Subcodes

BGP [RFC4271] Section 4.5 provides a number of subcodes to the NOTIFICATION message, and Section 6.2 elaborates on the use of those subcodes specific to OPEN message.

This specification introduces three new subcodes for OPEN Message Error code:

7 - Capability Value Mismatch - Session Id mismatch, i.e. remote speaker whishes to use different capability codes in Session Id compare to local speaker

8 - Grouping Conflict - values of capability codes used in Session Id of the received message cannot be unambiguously mapped to a locally configured group

9 - Grouping Required (from earlier drafts, perhaps should be removed if not used)

BGP implementations conforming to this specification SHOULD use new sub-codes as described further down in section "Connection establishment" of this document.

6. Modified Connection Collision Handling

BGP speaker conforming to and actively using this specification MUST

use modified connection collision handling procedure as described in this section.

Two sessions are said to collide if and only if both of following conditions are true:

1: the IP addresses on of peers are the same on both sessions

2: values of capability codes used in session identifier are either the same or overlapping (regardless fully or partially) within given capability code

Otherwise two sessions are considered unique and both MAY transition to the ESTABLISHED state (subject to rest of BGP specification).

Before attempting to create new session local system SHOULD evaluate existing sessions with the same peer. If there is already a session with the same peer in ESTABLISHED state and new session would collide with it, BGP speaker SHOULD NOT attempt creating new session; it's a good idea to notify operator of the local system about such potential collision.

Upon receipt of an OPEN messages BGP speaker MUST evaluate existing sessions with the same peer. If there is already a session in ESTABLISHED state and multisession distinguisher values of the old and the new OPEN messages fully match, the old session remains and the new MUST be closed.

If there is a session in OpenConfirm or OpenSent state and two sessions do not collide according to this document, then both sessions proceed as normally and section 6.8 of RFC4271 MUST NOT be applied. If on the other hand two sessions collide according to definition of this document, then original procedure from section 6.8 of RFC4271 MUST be applied, except for the NOTIFICATION type. Whereas original specification prescribes to use 'Cease' error code, multisession enabled BGP speaker SHOULD send NOTIFICATION message as described in this document.

7. Connection establishment

When BGP Multisession is enabled by configuration for given peer and configuration dictates that multiple sessions can potentially be established with given peer, BGP speaker MUST advertise Multisession Capability code in the OPEN message on every session with given peer. In all other cases Multisession capability SHOULD NOT be advertised. The value of Session Id MUST be the same on every session.

When Multisession-enabled BGP speaker receives an OPEN message without BGP Multisession Capability code it MUST assume that peer is not capable of multiple sessions and MUST use original Connection Collision Detection procedure as described in section 6.8 of RFC4271.

When Multisession-enabled BGP speaker receives an OPEN message containing BGP Multisession Capability Code but with Session Id not matching its own Session Id, local BGP speaker MUST send NOTIFICATION message with Error Code set to 2 ("OPEN Message Error") and Error Sub-code set to 8 ("Grouping Conflict") and drop the session. If received Session Id matches locally configured Session Id then BGP speaker MUST verify whether this session would collide with any of the existing as described in section "Modified Connection Collision Handling".

If session is allowed to continue by connection collision detection procedure, the next step for local speaker is to find matching group as follow:

1. If BGP capability code values used in Session Id of the received message match exactly (i.e. for every value in the received OPEN message there is corresponding value in a locally configured group) then local BGP speaker proceeds with this session

2. If values in the received message do not match any of the locally configured groups exactly, but there is one and only one locally configured group such that for every capability code the intersection between received and local values is non-empty set, then this group is selected for continuing the session. Note, such partial match results in behaviour similar to non-multisession BGP when capability codes overlap partially. Rationale behind allowing only one group for partial matching is that it simplifies specification and implementation; from operational perspective multiple partially matching groups suggest significant descrepancy in configuration between peers and therefore unlikely to be required in real-life networks.

3. In all other cases local BGP speaker MUST send NOTIFICATION message with Error Code set to 2 (OPEN Message Error) and Error Sub-code set to 8 (Grouping conflict).

Once local BGP speaker has identified which locally configured group corresponds to received OPEN message it proceeds with the session like it would have been regular non-multisession one, particularly - the original Finite State Machine applies. BGP speaker is free to handle such session either in the same process/thread as the one that received OPEN message, or it can hand over connection to another process/thread. If uses, the connection handover is local-matter of

BGP implementation and not part of this specification. Appendix contains an example how such handover could be done.

8. Graceful restart

With respect to Section 4.2 of BGP Graceful Restart [RFC4724], when determining whether a new connection BGP speaker evaluate values of all capability codes used in Session Identifier.

9. Error handling

If multisession-enabled BGP speaker detects an error condition that warrants session reset, it SHOULD reset only session that was affected by the error. Resetting other sessions with the same peer would significantly diminish value of multisession extensions.

10. Operational considerations

Multisession feature SHOULD be disabled by default. BGP implementation SHOULD provide configuration-time option to enable multisession extension on per-peer basis. If BGP implementation supports non-trivial groups, then it SHOULD provide configuration-time option for operator to control how sessions are grouped. An example of such option would be possibility for an operator to specify which address families will be carried in one session, and which address families will be carried in another session.

BGP implementation supporting multisession extension SHOULD allow operator to view state of each individual group and at least last NOTIFICATION message that caused connection reset.

For the sake of interoperability between BGP speakers supporting multisession, an implementation SHOULD NOT impose hard-coded restrictions on groups based on particular Session Id are put together. If such restrictions are unavoidable, then BGP implementation MUST support at least trivial groups based on that attribute. Let's consider this on an example. If implementation A requires AFI/SAFI 1/1 and 1/4 to be always carried within same session, while implementation B requires AFI/SAFI 1/4 to be always carried only with 1/128 and not with any other, then it's not possible to establish session between such BGP speakers. However if implementations A and B both allow each AFI/SAFI to be carried each in its own group, then we can establish three sessions - one for AFI/SAFI 1/1, another one for AFI/SAFI 1/4 and third one for AFI/SAFI 1/128.

11. Backward Compatibility

This subsection discusses a BGP speaker's behavior towards a peer that is known or assumed not to support this specification. In short, the BGP speaker's behavior towards such a peer should be as otherwise defined for the BGP protocol, according to [RFC4271] and any other extension supported by the BGP speaker.

If a BGP speaker receives OPEN message that doesn't include Multisession Capability and local BGP speaker is required to use multisession (e.g. through configuration by operator), the local BGP speaker MUST drop the session and send appropriate NOTIFICATION message as described in Section 5. If multisession is not required, local BGP speaker proceeds with multisession extension disabled, so it appears as regular implementation to the peer.

As previously mentioned, the BGP speaker SHOULD always advertise the Multisession capability in its OPEN message, even towards "backward compatibility" peers.

Use of techniques such as dynamic capabilities [I-D.ietf-idr-dynamic-cap] for on-the-fly switching of session modes is beyond the scope of this document.

12. State Machine

This specification does not modify BGP FSM as such, but all references to execution of collision handling procedure of original BGP specification are replaced with call to collision handling procedure described in this document.

The specific state machine modifications to [RFC4271] Section 8.2.2 are as follows.

13. Discussion

Note that many BGP implementations already permit multiple sessions to be used between a given pair of routers, typically by configuring multiple IP addresses on each router and configuring each session to be bound to a different IP address. The principal contribution of this specification is to allow multiple sessions to be created automatically, without additional configuration overhead or address consumption.

The specification supports the simple case of one capability being used as the session identifier and one connection per session

identifier value. It also permits connections be established based on multiple capabilities as a session identifier with multiple values per capability grouped together per connection.

In the context of MP-BGP based connections, which we believe may be the most prevalent use of this specification, it permits supporting one AFI/SAFI per connection, and also permits arbitrary grouping of AFI/SAFI onto BGP connections. For such grouping to function pleasingly, both peers participating in a connection need to agree on what AFI/SAFI groupings will be used. If conflicting groupings are configured, the connections may not establish, or more connections may be established than were expected (in the degenerate case, one connection per AFI/SAFI could be established despite configured groupings). We observe that the potential for misbehavior in the presence of conflicting configuration is not unusual in BGP, and that support for, and configuration of grouping is purely optional.

14. Security Considerations

This document does not change the BGP security model.

15. Acknowledgements

The authors would like to thank Martin Djernaes, Pedro Marques, Keyur Patel, Robert Raszuk, Yakov Rekhter, David Ward and Anton Elita for their valuable comments.

16. IANA Considerations

IANA has allocated BGP Capability Code 68 as the Multisession BGP Capability.

This document requests IANA to allocate three new OPEN Message Error subcodes:

7 - Capability Value Mismatch

8 - Grouping Conflict

9 - Grouping Required

17. References

17.1. Normative References

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

[RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

[RFC4724] Sangli, S., Chen, E., Fernando, R., Scudder, J., and Y. Rekhter, "Graceful Restart Mechanism for BGP", RFC 4724, January 2007.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, January 2007.

[RFC5492] Scudder, J. and R. Chandra, "Capabilities Advertisement with BGP-4", RFC 5492, February 2009.

17.2. Informative References

[I-D.ietf-idr-dynamic-cap]
Chen, E. and S. Ramachandra, "Dynamic Capability for BGP-4", draft-ietf-idr-dynamic-cap-12 (work in progress), October 2010.

[RFC2918] Chen, E., "Route Refresh Capability for BGP-4", RFC 2918, September 2000.

Appendix A. Multisession usage scenarios

This section demonstrates usage of Multisession Extension in several common scenarios. All examples presented here for illustrative purpose only, they're not part of Multisession specification.

A.1. Single session on both sides

BGP Speaker A and BGP Speaker B are both configured to exchange IPv4 unicast (AFI=1, SAFI=1) and IPv4 L3VPN (AFI=1, SAFI=128) prefixes over single session. If Multisession extension is disabled by configuration on both sides, then the session is, from every perspective, indistinguishable from ordinary (non-multisession) BGP peering. If only one of the speakers is enabled (through configuration) for multisession and yet only with one session to multiplex both AFI/SAFI, then again only single session is established and it looks like normal session. Although multisession-enabled BGP speaker is capable of processing new NOTIFICATION sub-

codes, the other side (non-multisession) won't take advantage of it. On the other hand use of new NOTIFICATION sub-codes isn't necessary in this situation because both sides keep all AFI/SAFI within same session. Finally, if both speakers are multisession-enabled, they still setup single session, but now they can use new NOTIFICATION sub-codes for more sophisticated error handling.

Note that if both speakers configured to use only single session and their respective AFI/SAFI lists overlap but do not match exactly, then like with ordinary (non-multisession) BGP speakers the session will transition to ESTABLISHED state. It's possible that one of the speakers (or both) require exact match of AFI/SAFI lists in order to establish session (either by implementation or through configuration). In this case such speaker will send NOTIFICATION message with Error Code 2 (OPEN Message Error) and Sub-code 8 (Grouping conflict) and subsequently close the session.

A.2. Single session on one side, multiple sessions on the other

In this setup Speaker A is configured to carry IPv4 unicast (AFI=1, SAFI=1) and IPv4 L3VPN (AFI=1, SAFI=128) prefixes within single session, while Speaker B is configured with two sessions - one for IPv4 unicast and second for IPv4 L3VPN. Several scenarios are possible depending on which speaker sends OPEN message first and whether Speaker A is multisession-enabled or not.

Assuming Speaker A is not multisession-enabled, it sends OPEN message first and there is no existing session between these two peers. Speaker B determines that OPEN message lists both AFI/SAFI and it knows that it wants to split them into different sessions, therefore it's obvious that setup cannot function as intended. Since separation of two address families into two groups is performed by operator (as per Multisession Extension specification), the most appropriate action is to prevent any communication between Speaker A and B until operator intervenes and resolves the conflict in configuration. To do this BGP Speaker B sends NOTIFICATION message with Error Code 6 (because peer is not expected to understand new notification sub-codes). Would Speaker A be multisession enabled, then Speaker B would send NOTIFICATION message with Error Code 1 and Error Subcode 9 (Grouping Required).

Now let's consider reverse situation - the Speaker B sends an OPEN message for either AFI/SAFI first. Assuming Speaker A is not multisession-enabled, it will accept OPEN message containing either AFI/SAFI and will reply with OPEN message containing both AFI/SAFI. Although session might transitions for a brief period to ESTABLISHED state, the Speaker B upon receipt of the OPEN message will detect misconfiguration and send NOTIFICATION message with Error Code 6 as

in previous paragraph. Would Speaker A be multisession-enabled, it could detect misconfiguration on its own and send NOTIFICATION message with Error Code 1 and Error Subcode 8 (Grouping conflict).

There is possibility that Speaker A opens one TCP connection and sends its OPEN message, and simultaneously Speaker B opens one or two TCP connection(s) and sends OPEN message on each of them. Since Speaker A is not multisession-enabled, it will invoke original collision detection procedure and will drop one of the sessions. Speaker B seeing NOTIFICATION message with Cease error code concludes that Speaker A is not multisession-capable and that setup prescribed by Speaker B's configuration cannot be achieved. Depending on implementation of Speaker B a session for one of the AFI/SAFI may progress to ESTABLISHED state, but Speaker B will inform operator about incompatible configuration.

It's also possible that initially Speaker B has been configured with only one AFI/SAFI, e.g. IPv4 unicast. The session between two peers would come up as described in previous subsection. Now suppose Speaker B is configured with additional session to carry IPv4 L3VPN prefixes. Since Speaker A does not have multiple sessions configured, it won't send another OPEN message as long as first session is in ESTABLISHED state. Therefore it's only possible that Speaker B will attempt establishing second connection and send new OPEN message containing only IPv4 L3VPN AFI/SAFI. If Speaker A is non-multisession enabled, it will drop second session sending NOTIFICATION message. From this Speaker B can conclude that configuration of two sides is incompatible, will stop attempting to bring up IPv4 L3VPN session and will notify operator. Already ESTABLISHED session may remain unaffected (subject to Speaker B implementation), just like with non-multisession speakers.

A.3. Multiple sessions based on AFI/SAFI

This is most common use of multisession extension is to separate prefixes based on AFI/SAFI. Note that use of AFI/SAFI based groups is denoted by empty Optional Data field in Multisession Capability, which is the same as in previous two sections. Grouping configuration is devised from the list of actually advertised AFI/SAFI lists (MP-BGP Capability). This will be demonstrated in following examples.

Let's consider BGP Speaker A and BGP Speaker B both configured to exchange IPv4 unicast, IPv4 labelled-unicast and IPv4 L3VPN prefixes each in its own session. We start with no existing sessions between these speakers. Speaker A (though roles can reverse) sends OPEN message in which among other capabilities it announces MP-BGP Capability for AFI=1 SAFI=1 and Multisession Capability with empty

optional data field. Speaker B upon receipt of such message finds that it expects to exchange IPv4 unicast with Speaker B in a dedicated session. It accepts connection and sends similar OPEN message to Speaker A. As there were no existing sessions, collision handling procedure is not invoked at this time. Next Speaker A (but again it could be Speaker B) starts new TCP connection to Speaker B and sends OPEN message with MP-BGP Capability for AFI=1 SAFI=4 and Multisession Capability with empty optional data field. Speaker B is willing to exchange IPv4 labelled-unicast too, but before accepting the proposal it executes collision detection procedure. Since AFI/ SAFI lists of the old (ESTABLISHED) and of the new sessions are different, the sessions don't collide and, sending OPEN message with AFI=1 SAFI=4, the Speaker B brings second session to ESTABLISHED state. In the same way third session, for AFI=1 SAFI=128, is brought up.

Note that similar behaviour will be also observed if two speakers send OPEN messages simultaneously - modified collision handling procedure, introduced by Multisession Extension specification, will mark sessions as unique based on the difference in Session Id (different AFI/SAFI lists). If Speaker A opens TCP connection and sends an OPEN message for either AFI/SAFI, and simultanously Speaker B opens TCP connection and send OPEN message for the same AFI/SAFI, then modified collision handling procedure will resolve the conflict just like original procedure would do in non-multisession environment. Yet modified collision handling procedure allows sessions with distinct Session Id's to coexist without affecting each other. This behaviour applies also to more complex cases where groups include more AFI/SAFI or based on different Capability Codes all together. For this reason collision handling is not discussed in remaining scenarios.

Now suppose Speaker A configuration is as above, but Speaker B is configured to combine labelled-unicast and L3VPN prefixes into the same session. IPv4 session is brought up as above. Next there are two possible alternatives. Either Speaker A sends OPEN message for one of the remaining sessions, to which Speaker B responds with NOTIFICATION message Error Code 2 and Error Subcode 8. Or Speaker B sends OPEN message for combined session including both of the remaining address families, to which Speaker A responds either with exactly the same NOTIFICATION message. At the end only IPv4 session remains in ESTABLISHED state, while two other address families require operator's intervention for configuring either Speaker A with combined session for labelled-unicast and L3VPN, or Speaker B for one session per AFI/SAFI. Note that if Speaker B would have used an implementation that requires that labelled-unicast and L3VPN address-families are combined into single session, then behaviour of each side would be exactly as above.

If Speaker A wouldn't have L3VPN configuration for Speaker B at all, then whether second session would progress to ESTABLISHED or not depends on whether configuration of either side requires exact match between groups (by default implementations expected to mimim original BGP behaviour which will bring overlapping AFI/SAFI up, but won't require exact match, but some implementation may provide configuration knob to require exact match).

Finally we look at the case where AFI/SAFI lists of different configured sessions overlap. Suppose Speaker A is configured with following groups: group 1 AFI=1 SAFI=1, group 2 AFI=1 SAFI=4 and SAFI=128, group 3 AFI=2 SAFI=4; and Speaker B is configured as: group 1 AFI=1 SAFI=1, group 2 AFI=1 SAFI=4, group 3 AFI=1 SAFI=128 and AFI=2 SAFI=4. For simplicity sake we assume that group 1 is brought up first. Both speakers behave as already described in previous case. Next let Speaker A to be the first to setup second TCP session and send OPEN message for group 2. Applying collision handling procedure as defined in Multisession specification Speaker B continues processing of received OPEN message. If Speaker B is configured for strict match between the groups, then it will detect incompatibility of AFI/SAFI list between the received message and its own configuration, therefore it will send NOTIFICATION message with Error Code 2 and Error Subcode 8. If on the other hand Speaker B allows partial overlapping of received and its own AFI lists (as regular BGP implementation would in absence of multisession), it will reply with OPEN message that lists AFI=1 SAFI=4 and session potentially progresses to ESTABLISHED state provided that Speaker A doesn't require exact match on AFI/SAFI list. Similar applies to the session 3 for the remaining AFI/SAFI. Note that configuration for exact or partial match between AFI/SAFI lists is the same for all sessions between given peers.

A.4. Multiple sessions based on arbitrary BGP Capabilities

Although grouping based on arbitrary attributes is the most comprehensive scenario, the behaviour of the BGP speakers is essentially the same as in case of AFI/SAFI based groups. However arbitrary groups do add extra complexity because BGP speakers need to consider not only values of single capability, but need to agree upon Capability Codes that constitute Session Id. Following example demonstrates behaviour of multisession-enabled BGP speakers in situation where Session Id on each side is based on different capabilities.

Let's suppose there is imaginery Capability Code X that denotes Experiment Id, and two speakers would like to exchange IPv4 unicast and L3VPN prefixes for two experiments. Speaker A would like to group prefixes into separate sessions based solely on Experiment Id

(so two sessions with two AFI/SAFI in each), while Speaker B would like to have separate session per experiment per AFI/SAFI (so four sessions with one AFI/SAFI in each). Since Session Id involves attribute other than AFI/SAFI, the Optional Data field in Multisession Capability will be non-empty. Multisession Capability sent by Speaker A will contain only 'Experiment Id Capability Code' in the Optional Data, whereas Speaker B will put there both "Experiment Id Capability Code" and "MP-BGP (AFI/SAFI)". When either speaker receives OPEN message from the peer, it will notice mismatch between content of the Optional Data field and, since sessions cannot be established as intended, the speaker will send NOTIFICATION message with Error Code 2 and Subcode 7 after which session will be dropped. Both speakers will notify operator and will suppress further attempt to bring session up until configuration of either side changes.

Note that despite Multisession Capability does not containing a field to denote support for non-AFI/SAFI based groups, even an implementation that does not support groups based on arbitrary capability codes will be able to recognise configuration mismatch and provide sufficient information to the peer as described above.

A.5. Process level separation of multiple sessions

As fault isolation is the key motivation for the Multisession Extension it's natural to consider process-level separation between the sessions. Although Multisession specification itself does not prescribe any particular way of handling each session, BGP implementations can leverege IPC facilities provided by host operating systems to handover arbitrary session to appropriate process. For example, many systems can pass connection from the process that accepted TCP connection to a process dedicated for particular group using specially crafted message on Unix socket. This is somewhat acking to inetd, but based on content of the OPEN message (e.g. AFI/SAFI list) rather than on transport protocol properties (e.g. TCP/UDP port numbers). At one extrimity the process that initially accepts TCP connection may be very primitive and can leave even connection collision handling to a specializing process, on the other hand process could handle collision detection itself or even handle particular group on its own while passing only specific group to another process. This process level separation is local implementation business and does not require specific aid from BGP at protocol specification level. Therefore process level separation is not part of multisession specification.

Authors' Addresses

John G. Scudder
Juniper Networks

Email: jgs@juniper.net

Chandra Appanna
Cisco Systems

Email: achandra@cisco.com

Ilya Varlashkin
Easynet Global Services

Email: ilya.varlashkin@easynet.net