

Workgroup: Internet Engineering Task Force  
Internet-Draft: draft-ietf-idr-long-lived-gr-05  
Updates: [6368](#) (if approved)  
Published: 29 June 2023  
Intended Status: Standards Track  
Expires: 31 December 2023  
Authors: J. Uttaro  
          Independent Contributor  
          B. Decraene  
          Orange  
          E. Chen  
          Palo Alto Networks  
          J. G. Scudder  
          Juniper Networks  
**Support for Long-lived BGP Graceful Restart**

## Abstract

In this document, we introduce a new BGP capability termed "Long-lived Graceful Restart Capability" so that stale routes can be retained for a longer time upon session failure than is provided for by BGP Graceful Restart (RFC 4724). A well-known BGP community "LLGR\_STALE" is introduced for marking stale routes retained for a longer time. A second well-known BGP community, "NO\_LLGR", is introduced to mark routes for which these procedures should not be applied. We also specify that such long-lived stale routes be treated as the least-preferred, and their advertisements be limited to BGP speakers that have advertised the new capability. Use of this extension is not advisable in all cases, and we provide guidelines to help determine if it is.

We update RFC 6368 by specifying that the LLGR\_STALE community must be propagated into, or out of, the path attributes exchanged between PE and CE.

## 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 31 December 2023.

## Copyright Notice

Copyright (c) 2023 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

- [1. Introduction](#)
    - [1.1. Requirements Language](#)
  - [2. Definitions](#)
  - [3. Protocol Extensions](#)
    - [3.1. Long-lived Graceful Restart Capability](#)
    - [3.2. LLGR STALE Community](#)
    - [3.3. NO LLGR Community](#)
  - [4. Theory of Operation](#)
    - [4.1. Use of Graceful Restart Capability](#)
    - [4.2. Session Resets](#)
    - [4.3. Processing LLGR STALE Routes](#)
    - [4.4. Route Selection](#)
    - [4.5. Errors](#)
    - [4.6. Optional Partial Deployment Procedure](#)
    - [4.7. Procedures when BGP is the PE-CE Protocol in a VPN](#)
      - [4.7.1. Procedures when EBGP is the PE-CE Protocol in a VPN](#)
      - [4.7.2. Procedures when IBGP is the PE-CE Protocol in a VPN](#)
  - [5. Deployment Considerations](#)
    - [5.1. When BGP is the PE-CE Protocol in a VPN](#)
    - [5.2. Risks of Depreferencing Routes](#)
  - [6. Security Considerations](#)
  - [7. Examples of Operation](#)
  - [8. Acknowledgements](#)
  - [9. Contributors](#)
  - [10. IANA Considerations](#)
  - [11. References](#)
    - [11.1. Normative References](#)
    - [11.2. Informative References](#)
- [Authors' Addresses](#)

## 1. Introduction

Historically, routing protocols in general, and BGP in particular, have been designed with a focus on correctness, where a key part of "correctness" is for each network element's forwarding state to converge toward the current state of the network as quickly as possible. For this reason, the protocol was designed to remove state advertised by routers that went down (from a BGP perspective) as quickly as possible. Over time, this has been relaxed somewhat, notably by BGP Graceful Restart (GR) [[RFC4724](#)]; however, the paradigm has remained one of attempting to rapidly remove "stale" state from the network.

Over time, two phenomena have arisen that call into question the underlying assumptions of this paradigm. The first is the widespread adoption of tunneled forwarding infrastructures, for example, MPLS. Such infrastructures eliminate the risk of some types of forwarding loops that can arise in hop-by-hop forwarding and thus reduce one of the motivations for strong consistency between forwarding elements. The second is the increasing use of BGP as a transport for data which is less closely associated with packet forwarding than was originally the case. Examples include the use of BGP for autodiscovery ([VPLS](#) [[RFC4761](#)]) and filter programming ([FLOWSPEC](#) [[RFC8955](#)]). In these cases, BGP data takes on a character more akin to configuration than to traditional routing.

The observations above motivate a desire to offer network operators the ability to choose to retain BGP data for a longer period than has hitherto been possible when the BGP control plane fails for some reason. Although the semantics of BGP Graceful Restart [[RFC4724](#)] are close to those desired, several gaps exist, most notably in the maximum time for which "stale" information can be retained -- Graceful Restart imposes a 4095-second upper bound.

In this document, we introduce a new BGP capability termed "Long-lived Graceful Restart Capability" so that stale information can be retained for a longer time across a session reset. We also introduce two new BGP well-known communities, "LLGR\_STALE", to mark such information, and "NO\_LLGR", to indicate that these procedures should not be applied to the marked route. Long-lived stale information is to be treated as least-preferred, and its advertisement limited to BGP speakers that support the new capability. Where possible, we reference the semantics of BGP Graceful Restart [[RFC4724](#)] rather than specifying similar semantics in this document.

The expected deployment model for this extension is that it will only be invoked for certain address families. This is discussed in more detail in [the Deployment Considerations section](#) ([Section 5](#)). When used, its use may be combined with that of traditional Graceful

Restart, in which case it is invoked only after the traditional Graceful Restart interval has elapsed, or it may be invoked immediately. Apart from the potential to greatly extend the timer, the most obvious difference between Long-Lived and traditional Graceful Restart is that in the Long-Lived version, routes are "depreferenced", that is, treated as least-preferred, whereas in the traditional version, route preference is not affected. The design choice to treat Long-Lived Stale routes as least-preferred was informed by the expectation that they might be retained for a (potentially) almost unbounded period of time, whereas in the traditional Graceful Restart case, stale routes are retained for only a brief interval. In the Graceful Restart case, the tradeoff between advertising new route status (at the cost of routing churn) and not advertising it (at the cost of suboptimal or incorrect route selection) is resolved in favor of not advertising. In the LLGR case, it is resolved in favor of advertising new state, and using stale information only as a last resort.

[Section 7](#) provides some simple examples illustrating the operation of this extension.

### 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. Definitions

**CE:** A Customer Edge router. [[RFC4364](#)]

**Depreference, Depreferenced:** A route is said to be depreferenced if it has its route selection preference reduced in reaction to some event.

**EoR:** Marker for End-of-RIB, defined in [[RFC4724](#)] Section 2.

**GR:** Abbreviation for "Graceful Restart" [[RFC4724](#)], also sometimes referred to herein as "conventional Graceful Restart" or "conventional GR" to distinguish it from the "Long-lived Graceful Restart" defined by this document.

**Helper:** Or "helper router". During Graceful Restart or Long-lived Graceful Restart, the router that detects a session failure and applies the listed procedures. [[RFC4724](#)] refers to this as the "receiving speaker".

**LLGR:** Abbreviation for "Long-lived Graceful Restart".

**LLST:** Abbreviation for "Long-lived Stale Time".

**PE:** A Provider Edge router. [[RFC4364](#)]

**Route:** We use "route" to mean any information encoded as a BGP NLRI and set of path attributes. As discussed above, the connection between such routes and the installation of forwarding state may be quite remote.

**VRF:** VPN Routing and Forwarding table. [[RFC4364](#)]

### 3. Protocol Extensions

A new BGP capability and two new BGP communities are introduced.

#### 3.1. Long-lived Graceful Restart Capability

The "Long-lived Graceful Restart Capability", or "LLGR Capability" (value: 71) is a BGP capability [[RFC5492](#)] that can be used by a BGP speaker to indicate its ability to preserve its state according to the procedures of this document. This capability **MUST** be advertised in conjunction with the Graceful Restart capability [[RFC4724](#)], see [the "Use of Graceful Restart Capability" section \(Section 4.1\)](#).

The capability value consists of zero or more tuples <AFI, SAFI, Flags, Long-lived Stale Time> as follows:

```
+-----+
| Address Family Identifier (16 bits) |
+-----+
| Subsequent Address Family Identifier (8 bits) |
+-----+
| Flags for Address Family (8 bits) |
+-----+
| Long-lived Stale Time (24 bits) |
+-----+
| ... |
+-----+
| Address Family Identifier (16 bits) |
+-----+
| Subsequent Address Family Identifier (8 bits) |
+-----+
| Flags for Address Family (8 bits) |
+-----+
| Long-lived Stale Time (24 bits) |
+-----+
```

The meaning of the fields are as follows:

Address Family Identifier (AFI), Subsequent Address Family Identifier (SAFI):

The AFI and SAFI, taken in combination, indicate that the BGP speaker has the ability to preserve its forwarding state for the address family during a subsequent BGP restart. Routes may be explicitly associated with a particular AFI and SAFI using the encoding of [\[RFC4760\]](#) or implicitly associated with <AFI=IPv4, SAFI=Unicast> if using the encoding of [\[RFC4271\]](#).

Flags for Address Family:

This field contains bit flags relating to routes that were advertised with the given AFI and SAFI.

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

The most significant bit is used to indicate whether the state for routes that were advertised with the given AFI and SAFI has indeed been preserved during the previous BGP restart. When set (value 1), the bit indicates that the state has been preserved. This bit is called the "F bit" since it was historically used to indicate the preservation of Forwarding State. Use of the F bit is detailed in the [Session Resets section \(Section 4.2\)](#). The remaining bits are reserved and MUST be set to zero by the sender and ignored by the receiver.

Long-lived Stale Time:

This time (in seconds) specifies how long stale information (for this AFI/SAFI) may be retained by the receiver (in addition with the period specified by the "Restart Time" in the Graceful Restart Capability). Because the potential use cases for this extension vary widely, there is no suggested default value for the LLST.

### 3.2. LLGR\_STALE Community

The well-known BGP community [\[RFC1997\]](#) "LLGR\_STALE" (value: 0xFFFF0006) can be used to mark stale routes retained for a longer period of time. Such long-lived stale routes are to be handled according to the procedures specified in the [Operation section \(Section 4\)](#).

An implementation MAY allow users to configure policies that accept, reject, or modify routes based on the presence or absence of this community.

### 3.3. NO\_LLGR Community

The well-known BGP community "NO\_LLGR" (value: 0xFFFF0007) can be used to mark routes that a BGP speaker does not want to be treated according to these procedures, as detailed in [the Operation section \(Section 4\)](#).

An implementation MAY allow users to configure policies that accept, reject, or modify routes based on the presence or absence of this community.

## 4. Theory of Operation

If A BGP speaker is configured to support the procedures of this document, it MUST use [BGP Capabilities Advertisement \[RFC5492\]](#) to advertise the "Long-lived Graceful Restart Capability". The setting of the parameters for an AFI/SAFI depends on the properties of the BGP speaker, network scale, and local configuration.

In the presence of the "Long-lived Graceful Restart Capability", the procedures specified in [\[RFC4724\]](#) and [\[RFC8538\]](#) continue to apply unless explicitly revised by this document.

### 4.1. Use of Graceful Restart Capability

The Graceful Restart capability MUST be advertised in conjunction with the LLGR capability. If it is not so advertised, the LLGR capability MUST be disregarded. The purpose for mandating that both be used in conjunction is to enable the reuse of certain base mechanisms that are common to both "flavors", notably origination, collection, and processing of EoR, as well as the finite state machine modifications and connection reset logic introduced by GR.

We observe that if support for conventional Graceful Restart is not desired for the session, the conventional GR phase can be skipped by omitting all AFI/SAFI from the GR capability, advertising a Restart Time of zero, or both. The [Session Resets section \(Section 4.2\)](#) discusses the interaction of conventional and long-lived GR.

### 4.2. Session Resets

[BGP Graceful Restart \[RFC4724\]](#), updated by [\[RFC8538\]](#), defines conditions under which a BGP session can reset and have its associated routes retained. If such a reset occurs for a session for which the LLGR Capability has also been exchanged, the following procedures apply.

If the Graceful Restart Capability that was received does not list all AFI/SAFI supported by the session, then for those non-listed AFI/SAFI the GR "Restart Time" shall be deemed zero. Similarly, if the

received LLGR Capability does not list all AFI/SAFI supported by the session, then for those non-listed AFI/SAFI the "Long-lived Stale Time" shall be deemed zero.

The following text in [Section 4.2 of the GR specification \[RFC4724\]](#) no longer applies:

If the session does not get re-established within the "Restart Time" that the peer advertised previously, the Receiving Speaker MUST delete all the stale routes from the peer that it is retaining.

and the following procedures are specified instead:

After the session goes down, and before the session is re-established, the stale routes for an AFI/SAFI MUST be retained. The interval for which they are retained is limited by the sum of the "Restart Time" in the received Graceful Restart Capability and the "Long-lived Stale Time" in the received Long-lived Graceful Restart Capability. These timers MAY be modified by local configuration.

If the value of the "Restart Time" or the "Long-lived Stale Time" is zero, the duration of the corresponding period would be zero seconds. For example, if the "Restart Time" is zero and the "Long-lived Stale Time" is nonzero, only the procedures particular to LLGR would apply. Conversely, if the "Long-lived Stale Time" is zero and the "Restart Time" is nonzero, only the procedures of GR would apply. If both are zero, none of these procedures would apply, only those of the base BGP specification (although EoR would still be used as detailed in [\[RFC4724\]](#)). And finally, if both are nonzero, then the procedures would be applied serially -- first those of GR, then those of LLGR. We observe that during the first interval, while the procedures of GR are in effect, route preference would not be affected. During the second interval, while LLGR procedures are in effect, routes would be treated as least-preferred as specified elsewhere in this document.

Once the "Restart Time" period ends (including the case that the "Restart Time" is zero), the LLGR period is said to have begun and the following procedures MUST be performed:

\*For each AFI/SAFI for which it has received a nonzero "Long-lived Stale Time", the helper router MUST start a timer for that "Long-lived Stale Time". If the timer for the "Long-lived Stale Time" for a given AFI/SAFI expires before the session is re-established, the helper MUST delete all stale routes of that AFI/SAFI from the neighbor that it is retaining.

\*The helper router MUST attach the LLGR\_STALE community to the stale routes being retained. Note that this requirement implies



that the routes would need to be readvertised, to disseminate the modified community.

\*If any of the routes from the peer have been marked with the NO\_LLGR community, either as sent by the peer, or as the result of a configured policy, they MUST NOT be retained, but MUST be removed as per the normal operation of [\[RFC4271\]](#).

\*The helper router MUST perform the procedures listed under [Section 4.3](#).

Once the session is re-established, the procedures specified in [\[RFC4724\]](#) apply for the stale routes irrespective of whether the stale routes are retained during the "Restart Time" period or the "Long-lived Stale Time" period. However, in the case of consecutive restarts, the previously marked stale routes MUST NOT be deleted before the timer for the "Long-lived Stale Time" expires.

Similarly to [\[RFC4724\]](#), once the session is re-established, if the F bit for a specific address family is not set in the newly received LLGR Capability, or if a specific address family is not included in the newly received LLGR Capability, or if the LLGR and accompanying GR Capability are not received in the re-established session at all, then the Helper MUST immediately remove all the stale routes from the peer that it is retaining for that address family.

If a "Long-lived Stale Time" timer is running for routes with a given AFI/SAFI received from a peer, it MUST NOT be updated (other than by manual operator intervention) until the peer has established and synchronized a new session. The session is termed "synchronized" for a given AFI/SAFI once the EoR for that AFI/SAFI has been received from the peer, or once the Selection\_Deferral\_Timer discussed in [\[RFC4724\]](#) expires.

The value of a "Long-lived Stale Time" in the capability received from a neighbor MAY be reduced by local configuration.

While the session is down, the expiration of a "Long-lived Stale Time" timer is treated analogously to the expiration of the "Restart Time" timer in Graceful Restart, other than applying only to the AFI/SAFI it accompanies. However, the timer continues to run once the session has re-established. The timer is neither stopped nor updated until EoR is received for the relevant AFI/SAFI from the peer. If the timer expires during synchronization with the peer, any stale routes that the peer has not refreshed, are removed. If the session subsequently resets prior to becoming synchronized, any remaining routes (for the AFI/SAFI whose LLST timer expired) MUST be removed immediately.

### 4.3. Processing LLGR\_STALE Routes

A BGP speaker that has advertised the "Long-lived Graceful Restart Capability" to a neighbor MUST perform the following upon receiving a route from that neighbor with the "LLGR\_STALE" community, or upon attaching the "LLGR\_STALE" community itself per [Section 4.2](#):

\*Treat the route as the least-preferred in route selection (see below). See the [Risks of Depreferencing Routes section \(Section 5.2\)](#) for a discussion of potential risks inherent in doing this.

\*The route SHOULD NOT be advertised to any neighbor from which the Long-lived Graceful Restart Capability has not been received. The exception is described in the [Optional Partial Deployment Procedure section \(Section 4.6\)](#). Note that this requirement implies that such routes should be withdrawn from any such neighbor.

\*The "LLGR\_STALE" community MUST NOT be removed when the route is further advertised.

### 4.4. Route Selection

A "least-preferred" route MUST be treated as less preferred than any other route that is not also least-preferred. When performing route selection between two routes both of which are least-preferred, normal tie-breaking applies. Note that this would only be expected to happen if the only routes available for selection were least-preferred -- in all other cases, such routes would have been eliminated from consideration.

### 4.5. Errors

If the LLGR capability is received without an accompanying GR capability, the LLGR capability MUST be ignored, that is, the implementation MUST behave as though no LLGR capability had been received.

### 4.6. Optional Partial Deployment Procedure

Ideally, all routers in an Autonomous System would support this specification before it was enabled. However, to facilitate incremental deployment, stale routes MAY be advertised to neighbors that have not advertised the Long-lived Graceful Restart Capability under the following conditions:

\*The neighbors MUST be internal (IBGP or Confederation) neighbors.

\*The NO\_EXPORT community [[RFC1997](#)] MUST be attached to the stale routes.

\*The stale routes MUST have their LOCAL\_PREF set to zero. See the [Risks of Depreferencing Routes section \(Section 5.2\)](#) for a discussion of potential risks inherent in doing this.

If this strategy for partial deployment is used, the network operator should set LOCAL\_PREF to zero for all long-lived stale routes throughout the Autonomous System. This trades off a small reduction in flexibility (ordering may not be preserved between competing long-lived stale routes) for consistency between routers that do, and do not, support this specification. Since the consistency of route selection can be important for preventing forwarding loops, the latter consideration dominates.

#### **4.7. Procedures when BGP is the PE-CE Protocol in a VPN**

##### **4.7.1. Procedures when EBGP is the PE-CE Protocol in a VPN**

In VPN deployments, for example [[RFC4364](#)], EBGP is often used as a PE-CE protocol. It may be a practical necessity in such deployments to accommodate interoperation with peer routers that cannot easily be upgraded to support specifications such as this one. This leads to a problem: in this specification, we take pains to ensure that "stale" routing information will not leak beyond the perimeter of routers that support these procedures so that it can be depreferenced as expected, and we provide [a workaround \(Section 4.6\)](#) for the case where one or more IBGP routers are not upgraded. However, in the VPN PE-CE case, the protocol in use is EBGP, and our workaround does not work since it relies on the use of LOCAL\_PREF, an IBGP-only path attribute.

We observe that the principal motivation for restricting the propagation of "stale" routing information is the desire to prevent it from spreading without limit once it exits the "safe" perimeter. We further observe that VPN deployments are typically topologically constrained, making this concern moot. For this reason, an implementation MAY advertise stale routes over a PE-CE session, when explicitly configured to do so. That is, the second rule listed in [Section 4.3](#) MAY be disregarded in such cases. All other rules continue to apply. Finally, if this exception is used, the implementation SHOULD by default attach the NO\_EXPORT community to the routes in question, as an additional protection against stale routes spreading without limit. Attachment of the NO\_EXPORT community MAY be disabled by explicit configuration, to accommodate exceptional cases.

See further discussion of using explicitly configured policy to mitigate this issue in [Section 5.1](#).

#### 4.7.2. Procedures when IBGP is the PE-CE Protocol in a VPN

If IBGP is used as the PE-CE protocol, following the procedures of [\[RFC6368\]](#), then when a PE router imports a VPN route that contains the ATTR\_SET attribute into a destination VRF and subsequently advertises that route to a CE router,

\*If the CE router does support the procedures of this document (in other words, if the CE router has advertised the LLGR Capability): In addition to including in the advertised route the path attributes derived from the ATTR\_SET as per [\[RFC6368\]](#), the PE router MUST also include the LLGR\_STALE community if it is present in the path attributes of the imported route, even if it is not present in the ATTR\_SET attribute.

\*If the CE router does not support the procedures of this document, then the optional procedures of [Section 4.6](#) MAY be followed, attaching the NO\_EXPORT community and setting the value of LOCAL\_PREF to zero, overriding the value found in the ATTR\_SET.

Similarly, when a PE router receives a route from a CE into its VRF and subsequently exports that route to a VPN address family,

\*If the PE router does support the procedures of this document (in other words, if the PE router has advertised the LLGR Capability): In addition to including in the VPN route the ATTR\_SET derived from the path attributes as per [\[RFC6368\]](#), the PE router MUST also include the LLGR\_STALE community in the VPN route if it is present in the path attributes of the route as received from the CE.

\*If the PE router does not support the procedures of this document, there exists no ideal solution. The CE could advertise a route with LLGR\_STALE, with the understanding that the LLGR\_STALE marking will only be honored by the provider network if appropriate policy configuration exists on the PE (see [Section 5.1](#)). It is at least guaranteed that LLGR\_STALE will be propagated when the route is propagated beyond the provider network. Or, the CE could refrain from advertising the LLGR\_STALE route to the incapable PE.

## 5. Deployment Considerations

The deployment considerations discussed in [\[RFC4724\]](#) apply to this document. In addition, network operators are cautioned to carefully consider the potential disadvantages of deploying these procedures for a given AFI/SAFI. Most notably, if used for an AFI/SAFI that conveys traditional reachability information, the use of a long-lived

stale route could result in a loss of connectivity for the covered prefix. This specification takes pains to mitigate this risk where possible, by making such routes least-preferred and by restricting the scope of such routes to routers that support these procedures (or, optionally, a single Autonomous System, see ["Optional Partial Deployment Procedure"](#) ([Section 4.6](#))). However, according to the normal rules of IP forwarding a stale more-specific route, that has no non-stale alternate paths available, will still be used instead of a non-stale less-specific route. Networks in which the deployment of these procedures would be especially concerning include those which do not use "tunneled" forwarding (in other words, those using traditional hop-by-hop forwarding).

Implementations MUST NOT enable these procedures by default. They MUST require affirmative configuration per AFI/SAFI in order to enable them.

The procedures of this document do not alter the route resolvability requirement of [Section 9.1.2.1](#) of [\[RFC4271\]](#). Because of this, it will commonly be the case that "stale" IBGP routes will only continue to be used if the router depicted in the next hop remains resolvable, even if its BGP component is down. Details of IGP fault-tolerance strategies are beyond the scope of this document. In addition to the foregoing, it may be advisable to check the viability of the next hop through other means, for example, [BFD](#) [\[RFC5880\]](#). This may be especially useful in cases where the next hop is known directly at the network layer, notably EBGP.

As discussed in this document, after a BGP session goes down and before the session is re-established, stale routes may be retained for up to two consecutive periods, controlled by the "Restart Time" and the "Long-lived Stale Time", respectively. During the first period routing churn would be prevented but with potential blackholing of traffic. During the second period potential blackholing of traffic may be reduced but routing churn would be visible throughout the network. The setting of the relevant parameters for a particular application should take into account the tradeoffs, the network dynamics, and potential failure scenarios. If needed, the first period can be bypassed either by local configuration or by setting the "Restart Time" in the Graceful Restart Capability to zero and/or not listing the AFI/SAFI in that Capability.

The setting of the F bit (and the "Forwarding State" bit of the accompanying GR capability) depends in part on deployment considerations. The F bit can be understood as an indication that the Helper should flush associated routes (if the bit is left clear). As discussed in [the Introduction](#) ([Section 1](#)), an important use case for LLGR is for routes that are more akin to configuration than to

traditional routing. For such routes, it may make sense to always set the F bit, regardless of other considerations. Likewise, for control-plane-only entities such as dedicated route reflectors, that do not participate in the forwarding plane, it makes sense to always set the F bit. Overall, the rule of thumb is that if loss of state on the restarting router can reasonably be expected to cause a forwarding loop or black hole, the F bit should be set scrupulously according to whether state has been retained. Specifics of when the F bit is, and is not, set are implementation-dependent and may also be controlled by configuration. Also, for every AFI/SAFI represented in the LLGR capability that is also represented in the GR capability, there will be two corresponding F bits -- the LLGR F bit and the GR F bit. If the LLGR F bit is set, the corresponding GR F bit should also be set, since to do otherwise would cause the state to be cleared on the Receiving Router per the normal rules of GR, violating the intent of the set LLGR bit.

### **5.1. When BGP is the PE-CE Protocol in a VPN**

As discussed in [Section 4.7](#), it may be necessary for a PE to advertise stale routes to a CE in some VPN deployments, even if the CE does not support this specification. In that case, the operator configuring their PE to advertise such routes should notify the operator of the CE receiving the routes, and the CE should be configured to depreferenciate the routes.

Similarly, it may be necessary for a CE to advertise stale routes to a PE, even if the PE does not support this specification. In that case, the operator configuring their CE to advertise such routes should notify the operator of the PE receiving the routes, and the PE should be configured to depreferenciate the routes.

Typical BGP implementations will be able to be configured to depreferenciate routes by matching on the LLGR\_STALE community and setting the LOCAL\_PREF for matching routes to zero, similar to the procedure described in [Section 4.6](#).

### **5.2. Risks of Depreferencing Routes**

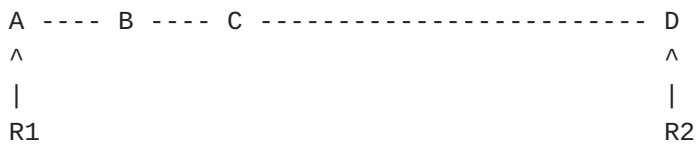
Depreferencing EBGP routes is considered safe, no different from the common practice of applying a routing policy to an EBGP session. However, the same is not always true of IBGP.

Consistent route selection is a fundamental tenet of IBGP correctness and safe operation in hop-by-hop routed networks. When routers within an AS apply different criteria in selecting routes, they can arrive at inconsistent route selections. This can lead to the formation of forwarding loops unless some form of tunneled forwarding is used to

prevent "core" routers from making a (potentially inconsistent) forwarding decision based on the IP header.

This specification uses the state of a peering session as an input to the selection criteria, depreferencing routes that are associated with a session that has gone down but have not yet aged out. Since different routers within an AS might have different notions as to whether their respective sessions with a given peer are up or down, they might apply different selection criteria to routes from that peer. This could result in a forwarding loop forming between such routers.

For an example of such a forwarding loop, consider the following simple topology:



In this example, A - D are routers with a full mesh of IBGP sessions between them (the sessions are not shown). The short links have unit cost, the long link has cost 5. Routers A and D are AS border routers, each advertising some route, R, into the AS -- these are denoted R1 and R2 in the diagram. In ordinary operation, it can be seen that routers B and C will select R1 for forwarding, and will forward toward A.

Suppose that the session between A and B goes down for some reason, and stays down long enough for LLGR processing to be invoked on B. Then on B, route R1 will be depreferenced, leading to the selection of R2 by B. However, C will continue to prefer R1. It can be seen that in this case, a forwarding loop for packets destined to R would form between B and C. (We note that other forwarding loop scenarios can be constructed for traditional GR, but are generally considered less severe since GR can remain in effect for a much more limited interval.)

The potential benefits of this specification can outweigh the risks discussed above, as long as care is exercised in deployment. The cardinal rule to be followed is, if a given set of routes are being used within an AS for hop-by-hop forwarding, it is not recommended to enable LLGR procedures. If tunneled forwarding (such as MPLS) is used within the AS, or if routes are being used for purposes other than hop-by-hop forwarding, less caution is needed, though the operator should still carefully consider the consequences of enabling LLGR.

## 6. Security Considerations

The security implications of the LLGR mechanism defined in this document are akin to those incurred by the maintenance of stale routing information within a network. This is particularly relevant when considering the maintenance of routing information that is used for service segregation - such as MPLS label entries.

For MPLS VPN services, the effectiveness of the traffic isolation between VPNs relies on the correctness of the MPLS labels between ingress and egress PEs. In particular, when an egress PE withdraws a label L1 allocated to a VPN1 route, this label MUST NOT be assigned to a VPN route of a different VPN until all ingress PEs stop using the old VPN1 route using L1.

Such a corner case may happen today if the propagation of VPN routes by BGP messages between PEs takes more time than the label re-allocation delay on a PE. Given that we can generally bound the worst-case BGP propagation time to a few minutes (for example 2-5), the security breach will not occur if PEs are designed to not reallocate a previously used and withdrawn label before a few minutes.

The problem is made worse with BGP GR between PEs as VPN routes can be stalled for a longer period of time (for example 20 minutes).

This is further aggravated by the BGP LLGR extension proposed in this document as VPN routes can be stalled for a much longer period of time (for example 2 hours, 1 day).

Therefore, to avoid VPN breach, before enabling BGP LLGR for a VPN address family, Service Providers need to check how fast a given label can be reused by a PE, taking into account:

- \*The load of the BGP route churn on a PE (in terms of the number of VPN labels advertised and the churn rate).

- \*The label allocation policy on the PE (possibly depending upon the size of the pool of the VPN labels (which can be restricted by hardware considerations or other MPLS usages), the label allocation scheme (for example per route or per VRF/CE), the re-allocation policy (for example least recently used label).

Note that [\[RFC4781\]](#) which defines Graceful Restart Mechanism for BGP with MPLS is also applicable to BGP LLGR.

These considerations notwithstanding, the LLGR mechanism described within this document is considered to be complex to exploit maliciously - in order to inject packets into a topology, there is a requirement to engineer a specific LLGR state between two PE devices,



whilst engineering label reallocation to occur in a manner that results in the two topologies overlapping. Such allocation is particularly difficult to engineer (since it is typically an internal mechanism of a router).

## 7. Examples of Operation

For illustrative purposes, we present a few examples of how this specification might be used in practice. These examples are neither exhaustive nor normative.

Consider the following scenario: A border router, ASBR1, has an IBGP peering with a route reflector, RR1, from which it learns routes. It has an EBGP peering with an external peer, EXT, to which it advertises those routes. The external peer has advertised the GR and LLGR Capabilities to ASBR1. ASBR1 is configured to support GR and LLGR on its sessions with RR1 and EXT. RR1 advertises a GR Restart Time of 1 (second) and an LLST of 3600 (seconds):

| Time     | Event   |
|----------|---|
| t        | ASBR1's IBGP session with RR fails. ASBR1 retains RR's routes according to the rules of <a href="#">GR [RFC4724]</a>  |
| t+1      | GR Restart Time expires. ASBR1 transitions RR's routes to long-lived stale by attaching the LLGR_STALE community and depreferencing them. However, since it has no backup routes, it continues to make use of them. It re-announces them to EXT with the LLGR_STALE community attached. |
| t+1+3600 | LLST expires. ASBR1 removes RR's stale routes from its own RIB and sends BGP updates to withdraw them from EXT.   |

Table 1

Next, imagine the same scenario but suppose RR1 advertised a GR Restart Time of zero, effectively disabling GR. Equally, ASBR1 could have used local configuration to override RR1's offered Restart Time, setting it to a locally-configured value of zero:

| Time     | Event  |
|----------|--|
| t        | ASBR1's IBGP session with RR fails. ASBR1 transitions RR's routes to long-lived stale by attaching the LLGR_STALE community and depreferencing them. However, since it has no backup routes, it continues to make use of them. It re-announces them to EXT with the LLGR_STALE community attached. |
| t+0+3600 | LLST expires. ASBR1 removes RR's stale routes from its own RIB and sends BGP updates to withdraw them from EXT.  |

Table 2

Next, imagine the original scenario, but consider that the ASBR1-RR1 session comes back up and becomes synchronized 180 seconds after the failure was detected:

| Time    | Event   |
|---------|---|
| t       | ASBR1's IBGP session with RR fails. ASBR1 retains RR's routes according to the rules of <a href="#">GR [RFC4724]</a>  |
| t+1     | GR Restart Time expires. ASBR1 transitions RR's routes to long-lived stale by attaching the LLGR_STALE community and depreferencing them. However, since it has no backup routes, it continues to make use of them. It re-announces them to EXT with the LLGR_STALE community attached. |
| t+1+179 | Session is reestablished and resynchronized. ASBR1 removes the LLGR_STALE community from RR1's routes and re-announces them to EXT with the LLGR_STALE community removed.   |

Table 3

Finally, imagine the original scenario, but consider that EXT has not advertised the LLGR Capability to ASBR1:

| Time     | Event   |
|----------|---|
| t        | ASBR1's IBGP session with RR fails. ASBR1 retains RR's routes according to the rules of <a href="#">GR [RFC4724]</a>  |
| t+1      | GR Restart Time expires. ASBR1 transitions RR's routes to long-lived stale by attaching the LLGR_STALE community and depreferencing them. However, since it has no backup routes, it continues to make use of them. It withdraws them from EXT. |
| t+1+3600 | LLST expires. ASBR1 removes RR's stale routes from its own RIB.   |

Table 4

## 8. Acknowledgements

We would like to thank Nabil Bitar, Martin Djernaes, Roberto Fragassi, Jeffrey Haas, Jakob Heitz, Daniam Henriques, Nicolai Leymann, Mike McBride, Paul Mattes, John Medamana, Pranav Mehta, Han Nguyen, Saikat Ray, and Bo Wu for their valuable input and contributions to the discussion and solution.

## 9. Contributors

Clarence Filsfils  
 Cisco Systems  
 Brussels 1150  
 Belgium

Email: cf@cisco.com

Pradosh Mohapatra  
Sproute Networks

Email: mpradosh@yahoo.com

Yakov Rekhter

Eric Rosen

Email: erosen52@gmail.com

Rob Shakir  
Google, Inc.  
1600 Amphitheatre Parkway  
Mountain View, CA 94043  
United States of America

Email: robjs@google.com

Adam Simpson  
Nokia

Email: adam.1.simpson@nokia.com

## 10. IANA Considerations

This document defines a new BGP capability - Long-lived Graceful Restart Capability. IANA has assigned a Capability Code of 71, from the "Capability Codes" registry.

This document introduces a new BGP well-known community "LLGR\_STALE" for marking long-lived stale routes, and another well-known community "NO\_LLGR" to mark routes that should not be retained if stale. IANA has assigned these well-known community values 0xFFFF0006 and 0xFFFF0007, respectively, from the "BGP Well-known Communities" registry.

For each of these three registrations, IANA is requested to update the reference to refer to this document.

IANA is requested to establish a new registry called "Long-lived Graceful Restart Flags for Address Family" under the Border Gateway Protocol (BGP) Parameters group. The registration procedures are Standards Action. The registry should initially be populated as follows:

| Bit Position | Name                  | Short Name | Reference     |
|--------------|-----------------------|------------|---------------|
| 0            | Preservation of state | F          | This document |
| 1-7          | Unassigned            |            |               |

Table 5

## 11. References

### 11.1. Normative References

- [RFC1997] Chandra, R., Traina, P., and T. Li, "BGP Communities Attribute", RFC 1997, DOI 10.17487/RFC1997, August 1996, <<https://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, <<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>>.
- [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>>.
- [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>>.
- [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>>.
- [RFC6368] Marques, P., Raszuk, R., Patel, K., Kumaki, K., and T. Yamagata, "Internal BGP as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 6368, DOI 10.17487/RFC6368, September 2011, <<https://www.rfc-editor.org/info/rfc6368>>.
- [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>>.
- [RFC8538] Patel, K., Fernando, R., Scudder, J., and J. Haas, "Notification Message Support for BGP Graceful Restart", RFC 8538, DOI 10.17487/RFC8538, March 2019, <<https://www.rfc-editor.org/info/rfc8538>>.

### 11.2. Informative References

- [RFC4364]

Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <<https://www.rfc-editor.org/info/rfc4364>>.

[RFC4761] Kompella, K., Ed. and Y. Rekhter, Ed., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, DOI 10.17487/RFC4761, January 2007, <<https://www.rfc-editor.org/info/rfc4761>>.

[RFC4781] Rekhter, Y. and R. Aggarwal, "Graceful Restart Mechanism for BGP with MPLS", RFC 4781, DOI 10.17487/RFC4781, January 2007, <<https://www.rfc-editor.org/info/rfc4781>>.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.

[RFC8955] Loibl, C., Hares, S., Raszuk, R., McPherson, D., and M. Bacher, "Dissemination of Flow Specification Rules", RFC 8955, DOI 10.17487/RFC8955, December 2020, <<https://www.rfc-editor.org/info/rfc8955>>.

#### Authors' Addresses

James Uttaro  
Independent Contributor

Email: [juttaro@ieee.org](mailto:juttaro@ieee.org)

Enke Chen  
Palo Alto Networks

Email: [enchen@paloaltonetworks.com](mailto:enchen@paloaltonetworks.com)

Bruno Decraene  
Orange

Email: [bruno.decraene@orange.com](mailto:bruno.decraene@orange.com)

John G. Scudder  
Juniper Networks

Email: [jgs@juniper.net](mailto:jgs@juniper.net)