Internet Engineering Task Force                  Naveen Anand
INTERNET-DRAFT                                    Janardhan
draft-janardhan-naveen-rtgwg-equalcostroutes-rip-00 27 June 2007
Expires:Dec 2007

Equal cost routes support for RIP/RIPNG

Status of this Document

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsolete by other documents at anytime. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
http://www.ietf.org/1id-abstracts.html
The list of Internet-Draft Shadow Directories can be accessed at
http://www.ietf.org/shadow.html

Abstract

RIP v2 as described in RFC 2453 and RIPNG as described in RFC 2080, do not support learning multiple equal cost routes to a destination through different neighbors. But in today’s internet it is very common to have topologies where there are more than one route to a destination. This document describes the modifications to the RIP/RIPNG protocols to handle this scenario and describes how to apply the split horizon and poison reverse for such routes. It is simple, light and is compatible with the implementations which do not support this feature.
### TABLE OF CONTENTS

1. Justification ..................................................3
2. Prerequisites ..................................................3
3. Overview ........................................................3
4. Processing of response messages .................................4
5. How to apply split horizon/poison reverse for equal cost routes? ....................................................6
6. How the router advertises equal cost routes in general? ......7
7. How routes should be advertised when Requests come from NON-RIP port? .................................................8
8. Equal cost support for RIPNG ......................................8
9. Contributors .......................................................8
10. References ........................................................8
11. Authors Address ................................................8
12. Copyright (C) The IETF Trust (2007) .............................9
1. Justification

The main motivation behind this draft is to document rules for RIP/RIPNG to handle equal cost routes to a destination, since most of the vendors already support this feature. But there is no concord on how to apply the split horizon/poison checks for such routes. In fact, most of the popular vendors apply the split/poison checks in a wrong way. This draft explicates the proper implementation of the above features. Also, since RIPNG is getting deployed, these features can be included while developing RIPNG.

2. Prerequisites

This draft assumes that, reader has familiarity with RIP and RIPNG protocols, as described in the RFCs, 2453 and 2080 respectively. For the sake of brevity we have not described many of the words and concepts which are already explained in the RIP/RIPNG rfcs.

3. Overview

In the following sections, we will use RIP v2 as reference while describing these features. The same holds good for RIPNG as well, unless otherwise mentioned.

RIPv2 RFC2453 predicates that RIP can support only one route to a destination. This is documented in the section 3.9.2 of RFC 2453. It states that, when a route is learnt to a destination, and if another route to that destination already exists in its routing table with the same metric, then new route can be ignored or route can be replaced with new one if the old route is showing signs of aging out. A strict implementation of the RFC will learn only one route a destination.

In today’s internet however, topologies normally have more than one route to a destination, hence it is useful if the rip routers can learn equal cost routes. This not only helps to increase the robustness but can also help in distributing the traffic among load balanced routers. Fortunately, it is simple to support this feature. Only modifications that need to be done in the old implementation are to change the response message processing (for supporting equal cost routes) and how split horizon/poison is applied for such routes (While advertising the routes to neighbors). And more importantly it is perfectly compatible with older implementations. Below we will discuss these features in detail.
4. Processing of response messages

This section is a emended version of the section 3.9.2 of RFC 2453, to support equal cost routes. It will be useful if the reader goes through the RFC before reading this.

A Response can be received for one of several different reasons:
- response to a specific query
- regular update (unsolicited response)
- triggered update caused by a route change

Processing is the same no matter why the Response was generated. Because processing of a Response may update the router’s routing table, the Response must be checked carefully for validity. The Response must be ignored if it is not from the RIP port. The datagram’s IPv4 source address should be checked to see whether the datagram is from a valid neighbor; the source of the datagram must be on a directly-connected network. It is also worth checking to see whether the response is from one of the router’s own addresses. Interfaces on broadcast networks may receive copies of their own broadcasts/multicasts immediately. If a router processes its own output as new input, confusion is likely so such datagrams must be ignored.

Once the datagram as a whole has been validated, process the RTEs in the Response one by one. Again, start by doing validation. Incorrect metrics and other format errors usually indicate misbehaving neighbors and should probably be brought to the administrator’s attention. For example, if the metric is greater than infinity, ignore the entry but log the event. The basic validation tests are:

- is the destination address valid (e.g., unicast; not net 0 or 127)
- is the metric valid (i.e., between 1 and 16, inclusive)

If any check fails, ignore that entry and proceed to the next. Again, logging the error is probably a good idea.

Once the entry has been validated, update the metric by adding the cost of the network on which the message arrived. If the result is greater than infinity, use infinity. That is,

\[
\text{metric} = \min (\text{metric} + \text{cost}, \infty)
\]
Now, check to see whether there is already an explicit route for the destination address. If there is no such route, add this route to the routing table, unless the metric is infinity (there is no point in adding a route which is unusable). Adding a route to the routing table consists of:

- Setting the destination address to the destination address in the RTE
- Setting the metric to the newly calculated metric (as described above)
- Set the next hop address to be the address of the router from which the datagram came
- Initialize the timeout for the route. If the garbage-collection timer is running for this route, stop it (see section 3.6 for a discussion of the timers)
- Set the route change flag
- Signal the output process to trigger an update (see section 3.8.1)

If there is an existing route, compare the next hop address to the address of the router from which the datagram came. If this datagram is from the same router as the existing route, reinitialize the timeout. Next, compare the metrics. If the datagram is from the same router as the existing route, and the new metric is different than the old one; or, if the new metric is lower than the old one; do the following actions:

- Adopt the route from the datagram (i.e., put the new metric in and adjust the next hop address, if necessary).
- Set the route change flag and signal the output process to trigger an update
- If the new metric is infinity, start the deletion process (described above); otherwise, re-initialize the timeout (age timer)
- If there are any equal cost routes then mark all the old equal cost routes for deletion.

If the new metric is infinity, the deletion process begins for the route, which is no longer used for routing packets. Note that the deletion process is started only when the metric is first set to infinity. If the metric was already infinity, then a new deletion process is not started.
If the new metric is the same as the old one, this should be treated as an equal cost route and a new route should be installed in the router's routing table. In this case, the forwarding engine may do load balancing of the traffic.

[NOTE : When a route is learnt to a destination and it has metric Less than previously existing equal cost routes, then all the equal cost routes should be marked for deletion, and new route should be used. Also, if a route is learnt to a destination and it has metric more than previously existing equal cost routes, then this route should be marked for deletion]

5. How to apply split horizon/poison reverse for equal cost routes?

The simple split horizon check as described in section3.4.3 of RFC 2453 is as follows:

The "simple split horizon" scheme omits routes learned from one neighbor in updates sent to that neighbor. "Split horizon with poisoned reverse" includes such routes in updates, but sets their metrics to infinity.

When a router has 2 or more routes to the same destination, how should it apply split horizon check to them? The simple split horizon as explained in the RFC 2453 asks not to send the route back onto a link from which it is learnt. But in this case, router may have routes to same destination learnt on different link. So, the application of the split horizon check as described in RFC2453 will lead to the advertisement of a route learnt on one link, to another link on which it has learnt a different equal cost route to the same destination.

Consider the above topology . In this case, router C will learn 2 routes to destination D, one through router A and other through router B. When C sends a periodic update on the link towards B,
it will include the route learnt to D through A. But this not correct, since what it is advertising is the same information it has learnt from B. Even though route is learnt on a different interface with respect to C, with respect to router B, it is as if C is not doing split/poison check and simply advertising back the route which B had sent to C. Obviously this will lead to the same problems which split horizon of routes wanted to solve.

The solution is not to advertise such routes. It means, never advertise equal cost routes on a link, if router has learnt a route to that destination from that link. This is termed as, Âsplit horizon per destinationÂ. Key to note is, split horizon is not done per route. It is done per destination. New split horizon and poison reverse checks can be stated like this.

The "split horizon per destination" scheme omits all the equal cost routes learnt for a destination, if it has a route to that destination from same neighbor to which updates are sent. "Split horizon with poisoned reverse per destination" includes only one route to that destination, but sets their metrics to infinity.

When poison reverse is done, only one route is should be advertised with cost as 16, no matter how many equal cost routes router learns. Taking the above example, C will advertise a single route with cost 16 on the links towards A and B.

6. How the router advertises equal cost routes in general?

When a router sends routing updates on a link, it has to do split/Poison check for routes as described above. If the route can be advertised after applying the "split/poison per destination" check, then, only one should be advertised. This is obvious since, sending all the equal cost routes, serves nothing more than filling the packet.

[NOTE: In some scenarios, the equal cost routes can have nexthops different from advertising router itself. In this scenario, it can advertise more than one routes to a destination on a link.]

When one of the equal cost routes ages out or becomes unreachable (Cost becomes INFINITY), then router may send a triggered update of the other routes it has to the destination.
7. How routes should be advertised when route requests come from NON-RIP port?

The request message from non-rip port is normally directed towards a particular router and is normally used for administrative and debugging purposes. So, all equal cost routes can be advertised to such queries by appropriately filling the nexthops.

8. Equal cost support for RIPNG

The above mentioned processing also applies to RIPNG. So, all the RIPNG routers should do the same processing, should apply the split and poison checks as described above.

9. Contributors

Akash Shetty, Prashanth John, Sanoj KV, Saravana, Jayanth Chennamangalam, Archana Patel, Satish Hampali, Vijayananda, Nanda Kishore, JGS Prasad, Parag Kaneria, Vinayak Bhat and Geethamala have contributed to the this idea.

10. References


11. Author’s address

Janardhan Kulkarni,
Citrix systems,
Banglore.
email: janardhan.kulkarni@citrix.com

Naveen Anand,
Huawei Technologies,
Banglore.
email: naveenanand@huawei.com
Copyright (C) The IETF Trust (2007)

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.