Network Working Group
Request for Comments: 2991
Category: Informational

D. Thaler
Microsoft
C. Hopps
NextHop Technologies

November 2000

Multipath Issues in Unicast and Multicast Next-Hop Selection

Status of this Memo

Copyright Notice

Abstract

Various routing protocols, including Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (ISIS), explicitly allow "Equal-Cost Multipath" (ECMP) routing. Some router implementations also allow equal-cost multipath usage with RIP and other routing protocols. The effect of multipath routing on a forwarder is that the forwarder potentially has several next-hops for any given destination and must use some method to choose which next-hop should be used for a given data packet.

1. Introduction

Various routing protocols, including OSPF and ISIS, explicitly allow "Equal-Cost Multipath" routing. Some router implementations also allow equal-cost multipath usage with RIP and other routing protocols. Using equal-cost multipath means that if multiple equal-cost routes to the same destination exist, they can be discovered and used to provide load balancing among redundant paths.

The effect of multipath routing on a forwarder is that the forwarder potentially has several next-hops for any given destination and must use some method to choose which next-hop should be used for a given data packet. This memo summarizes current practices, problems, and solutions.

2. Concerns

Several router implementations allow multipath forwarding. This is sometimes done naively via round-robin, where each packet matching a given destination route is forwarded using the subsequent next-hop, in a round-robin fashion. This does provide a form of load balancing, but there are several problems with approaches such as round-robin or random:

Variable Path MTU
: Since each of the redundant paths may have a different MTU, this means that the overall path MTU can change on a packet-by-packet basis, negating the usefulness of path MTU discovery.

Variable Latencies
: Since each of the redundant paths may have a different latency involved, having packets take separate paths can cause packets to always arrive out of order, increasing delivery latency and buffering requirements.

  Packet reordering causes TCP to believe that loss has taken place when packets with higher sequence numbers arrive before an earlier one. When three or more packets are received before a "late" packet, TCP enters a mode called "fast-retransmit" [6] which consumes extra bandwidth (which could potentially cause more loss, decreasing throughput) as it attempts to unnecessarily retransmit the delayed packet(s). Hence, reordering can be detrimental to network performance.

Debugging
: Common debugging utilities such as ping and traceroute are much less reliable in the presence of multiple paths and may even present completely wrong results.

In multicast routing, the problem with multiple paths is that multicast routing protocols prevent loops and duplicates by constructing a single tree to all receivers of the same group address. Multicast routing protocols deployed today (DVMRP, PIM-DM, PIM-SM) [2] construct shortest-path trees rooted at either the source, or another router known as a Core or Rendezvous Point. Hence, the way they ensure that duplicates will not arise is that a given tree must use only a single next-hop towards the root of the tree.

3. Requirements

In the remainder of this document, we will use the term "flow" to represent the granularity at which the router keeps state (if at all) for classes of traffic. The exact definition of a flow may depend on the actual implementation. For example, a flow might be identified solely by destination address, or it might be identified by (source address, destination address, protocol id) triplet. Hence "flow" is not necessarily synonymous with the term "microflow" as used in RFC 2474 [7], which also includes port numbers. Indeed, including transport-layer information in the next-hop selection process can actually be problematic. For example, if packets are fragmented, the transport-layer information may not be available in every packet. Furthermore, having the choice of path depend on transport-layer fields may negate the benefit of caching information such as MTU for use in subsequent connections between the same endpoints.

All of the problems outlined in the previous section arise when packets in the same unicast or multicast "flow" are split among multiple paths. The natural solution is therefore to ensure that packets for the same flow always use the same path.

Two additional features are desirable:

Minimal disruption
: When multipath is used, meaning that multiple routes contribute valid next-hops, the chances are higher of routes being added and deleted from consideration than when only the "best" route is used (in which case metric changes in alternate routes have no effect on traffic paths). Since a higher number of routes may actually be used for forwarding when multipath is in use, the potential for packet reordering and packet loss due to route flaps can be much greater than when not using multipath. Hence, it is desirable to minimize the number of active flows affected by the addition or deletion of another next-hop.

Fast implementation
: The amount of additional computation required to forward a packet should be small. For example, when doing round-robin, this computation might consist of incrementing (modulo the number of next-hops) a next-hop index.

4. Solutions

We now provide three possible methods for improving the performance of multipath and then discuss their applicability to unicast and multicast forwarding.

Modulo-N Hash
: To select a next-hop from the list of N next-hops, the router performs a modulo-N hash over the packet header fields that identify a flow. This has the advantage of being fast, at the expense of (N-1)/N of all flows changing paths whenever a next-hop is added or removed.

Hash-Threshold
: The router first selects a key by performing a hash over the packet header fields that identify the flow. The N next-hops have been assigned unique regions in the hash function's output space. By comparing the hash value against region boundaries the router can determine which region the hash value belongs to and thus which next-hop to use. This method has the advantage of only affecting flows near the region boundaries (or thresholds) when next-hops are added or removed. For ECMP hash-threshold's lookup can be done with a simple division (hash_value / fixed_region_size). When a next-hop is added or removed, between 1/4 and 1/2 of all flows change paths. An analysis of this method can be found in [3].

Highest Random Weight (HRW)
: The router computes a key for EACH next-hop by performing a hash over the packet header fields that identify the flow, as well as over the address of the next-hop. The router then chooses the next-hop with the highest resulting key value [4]. This has the advantage of minimizing the number of flows affected by a next-hop addition or deletion (only 1/N of them), but is approximately N times as expensive as a modulo-N hash.

The applicability of these three alternatives depends on (at least) two factors: whether the forwarder maintains per-flow state, and how precious CPU is to a multipath forwarder.

Some routers may maintain per-flow state for reasons other than for supporting multipath. For example, routers typically keep per-flow state for multicast flows so that they can maintain the list of interfaces to which packets in the flow should be copied.

If per-flow state is maintained in a multipath forwarder, then computation of the next-hop can be done by the router at state creation time. This entails no additional computations at packet forwarding time compared with normal forwarding to a single next-hop, since the next-hop is precomputed. In this case, any method can be used, including round-robin, random, modulo-N, hash-threshold or HRW. Hash functions such as modulo-N, hash-threshold and HRW are better if the forwarder state may be deleted for any reason during the lifetime of a flow since subsequent next-hop computations by the router will

always select the same path. This also improves the usefulness of debugging utilities such as traceroute. Finally, to maximize the stability of paths (and hence the usefulness of traceroute, etc.), the use of HRW is recommended over the other methods mentioned herein.

If per-flow state is not maintained by the forwarder, then using multiple next-hops requires that the next-hop be calculated at packet arrival time. When CPU is more precious than stability of flow paths, hash-threshold is recommended over the other methods mentioned herein.

4.1. Unicast Forwarding

Depending on the implementation, unicast forwarding may or may not keep per-flow state. We recommend that where forwarder implementations keep flow state, routers should use HRW at state creation time (and next-hop deletion time) to select the next-hop, and that forwarders without per-flow state use hash-threshold.

4.2. Multicast Forwarding

Today's multicast forwarding engines use a cache of forwarding entries indexed by group (or group prefix) and source (or source prefix). This means that today's multicast forwarder's always keep per-flow state, although for some multicast routing protocols, the "flow" may be fairly coarse (e.g., traffic from all sources to the same destination). Since per-flow state is kept by the forwarder, it is recommended that the router always use HRW to select the next-hop.

Routers using explicit-joining protocols such as PIM-SM [5] should thus use the multipath information when determining to which neighbor a join message should be sent. For example, when multiple next-hops exist for a given Rendezvous Point (RP) toward which a (*,G) Join should be sent, it is recommended that HRW be used to select the next-hop to use for each group.

5. Applicability

The algorithms discussed above (except round-robin) all rely on some form of hash function. Equal flow distribution is achieved when the hash function is uniformly distributed. Since the commonly used hash functions only become uniformly distributed when the number of inputs is relatively large, these algorithms are more applicable to routers used to route many flows, than in, for example, a small business setting.

6. Redundant Parallel Links

A related problem occurs when multiple parallel links are used between the same pair of routers. A common solution is to bundle the two links together into a "super"-link when is then used for routing. For multicast forwarding, this results in the two links being reduced to a single next-hop (over the combined link) which can be used to prevent duplicates. When a unicast or multicast packet is queued to the combined link, some method, such as those discussed earlier, is still required to determine the physical link on which to transmit the packet. If the parallel links are identical, then most of the concerns discussed in this document are avoided with the combined link. The exception is packet reordering, which can still occur with round-robin, adversely affecting TCP.

7. Security Considerations

This document discusses issues with various methods of choosing a next-hop from among multiple valid next-hops. As such, it does not directly impact the security of the Internet infrastructure or its applications.

One issue that is worth mentioning, however, is that when next-hop selection is predictable, an attacker can synthesize traffic that will all hash the same, making it possible to launch a denial-of-service attack that overloads a particular path. Since a special case of this is when the same (single) next-hop is always selected, such an attack is easiest when multipath is not being used. Introducing multipath routing can make such an attack more difficult; the more unpredictable the hash is, the harder it becomes to conduct a denial-of-service attack against any single link.

8. References

[1] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.

[2] Maufer, T., "Deploying IP Multicast in the Enterprise", Prentice-Hall, 1998.

[3] Hopps, C., "Analysis of an Equal-Cost Multi-Path Algorithm", RFC 2992, November 2000.

[4] Thaler, D., and C.V. Ravishankar, "Using Name-Based Mappings to Increase Hit Rates", IEEE/ACM Transactions on Networking, February 1998.

[5] Estrin, D., Farinacci, D., Helmy, A., Thaler, D., Deering, S., Handley, M., Jacobson, V., Liu, C., Sharma, P. and L. Wei, "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification", RFC 2362, June 1998.

[6] Allman, M., Paxson, V. and W. Stevens, "TCP Congestion Control", RFC 2581, April 1999.

[7] Nichols, K., Blake, S., Baker, F. and D. Black., "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

9. Authors' Addresses

Dave Thaler
Microsoft
One Microsoft Way
Redmond, WA 98052

Phone: +1 425 703 8835
EMail: dthaler@dthaler.microsoft.com

Christian E. Hopps
NextHop Technologies, Inc.
517 W. William Street
Ann Arbor, MI 48103-4943
U.S.A

Phone: +1 734 936 0291
EMail: chopps@nexthop.com

10. Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.