Network Working Group
Internet Draft
Intended Status: Informational
Created: July 13, 2008
Expires: January 12, 2009

D. Papadimitriou
J. Lowe
Alcatel-Lucent

Routing System Stability

draft-dimitri-grow-rss-02.txt

Status of this Memo

Copyright Notice

Abstract

Understanding the dynamics of the Internet routing system is fundamental to ensure its robustness/stability and to improve the mechanisms of the BGP routing protocol. This documents outlines a program of activity for identifying, documenting and analyzing the dynamic properties of the Internet and its routing system.

Conventions used in this document

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 [RFC2119].

Document History

This is the initial version of this document.

1. Introduction

Understanding the dynamics of the Internet routing system is fundamental to ensuring its stability and improving the mechanisms of the BGP routing protocol [RFC4271]. Investigations on the Internet routing system dynamics involve investigations on routing engine resource consumption, in particular, memory and CPU.

System resource consumption depends on two items. First, there is the size of the routing space. The greater the number of routing entries there are, the greater the memory requirement on a routing device, and the greater the need for increased processing and searching capabilities to perform lookup operations. Second, the greater the number of adjacency and peering relationships between routing devices, the greater the dynamics associated with the routing information updates exchanged between all these adjacencies and peerings. This activity also increases the memory requirements for the operation of the routing protocol.

In other words, as the routing system grows [Huston07a], so do the requirements for routing engine memory and processing capacity. From a routing dynamics viewpoint, minimizing the amount of BGP routing information exchanged by routers is key to grappling with increasing requirements on memory and CPU.

So, although current routing engines could potentially support up to O(1M) routing table entries instabilities resulting i) from routing protocol behavior, ii) routing protocol information exchanges, and iii) changes in network topology may adversely affect the network's ability to remain in a useable state for extended periods of time. Note however that in terms of number of active routing entries, such routing engine could at worst have to deal with O(1M) routes within the next 5 years, see [Fuller07].

2. Objectives

The overall goal is to identify, root cause and document - in a structured manner - occurrences of Internet routing stability phenomena using data from operational networks.

To help accomplish this goal, the following tasks will be undertaken.

1. Development of a methodology to process and interpret routing table data. One guiding principle will be to be able to reproduce phenomena previously observed at different locations. This work will include documenting what information to collect and how it should be archived.

2. Identification of a set of stability criteria and development of methods for using them to provide a better understanding of the routing system's stability. Other working groups may find this beneficial in addition to the GROW working group.

3. Begin investigation into how routing protocol behavior and network dynamics mutually influence each other. The nature of the observations collected in the first task will suggest directions to proceed with this work.

This proposed approach would allow rigor and consistency to be brought to the study of network and routing stability. For example, it would allow for a unified approach to the cross-validation of techniques for looking at improving path exploration effects on the routing system.

3. Relevance to the GROW working group charter

This effort fits into the GROW working group's charter to deal with BGP operational issues related to routing table growth rates and the dynamic properties of the routing system.

GROW has an advisory role to the IDR working group to provide commentary on whether BGP is addressing relevant operational needs and, where appropriate, suggest course corrections, which puts this effort in a central place in the BGP investigation process.

Also, since the GROW working group community is directly linked to the broader BGP operational community, this effort goes together with obtaining routing table data from the field.

4. Routing system stability

In order to begin the discussion defined in work item detailed in Section 2, point 2, this section proposes a number of definitions for common routing and network stability terms.

The stability of a routing system is characterized by its response (in terms of processing routing information) to inputs of finite amplitude.

These inputs may be classified as either internal system events, such as routing protocol configuration changes, or as external system events, such as routing information updates. Such events are sometimes loosely referred to as routing "instabilities"; however, this term should be reserved for discussion about how the routing system responds to such events.

A routing system, which returns to its initial equilibrium state, when disturbed by an external and/or internal event, is considered to be stable.

A routing system, which transitions to a new equilibrium state, when disturbed by an external and/or internal event, is considered to be marginally stable.

Such state transitions, whether stable or marginal, should occur before the arrival of new input events.

The magnitude of the output of a stable routing system is small whenever the input is small. That is, a single routing information update shall not result in output amplification. Equivalently, a stable system's output will always decrease to zero whenever the input events stop.

A routing system, which remains in an unending condition of transition from one state to another when disturbed by an external or internal event, is considered to be unstable.

The degree to which a routing system, or components thereof, can function correctly in the presence of input events is a measure of the robustness of the system.

A precise definition of stability requires the specification of the following elements:

o) The system being examined: for example, a system might be comprised of: the routing system and associated events, such as input events, outputs, and related arrival rates.

o) A convergence metric: a metric to define the convergence characteristics of the system.

o) A stability metric: a metric that describes the degree of stability of the system and indicates how close the system is to being unstable.

The convergence and stability metrics may be affected by the following parameters:

o) The number of routing entries (where, each entry R toward an existing prefix D has an associated attribute set A consisting of AS-Path, MED, and Local Preference, etc.);

o) The number of CPU cycles, C, required to process a routing entry, and its associated memory space, M;

o) The input events and their arrival rates;

o) The output events associated with the processing of each input event.

5. Mathematical formulation

Section 4 outlined some proposals for definitions of commonly used stability terms applied to network and routing systems. In this section, an initial attempt is made to build a mathematical formulation around those concepts in order to begin the development of more practical metrics.

5.1 General Formulation

Let RT be the "Routing Table" and RT(n) represent the routing table at some time n. At time n+1, the routing table can be expressed as the sum of two components:

RT(n+1) = RTo(n) + deltaRT(n+1) (1)

In this equation, RTo(n) is the set of routes that experience no change between n and n+1, and deltaRT(n+1) accounts for all route changes (additions, deletions, and changes to previously existing routes) between n and n+1. deltaRT(n+1) itself can expressed as the sum of two components:

deltaRT(n+1) = RTc(n+1) + RTn(n+1) (2)

In this equation, RTc(n+1) is a set of routes at time n that experience some sort of change at time n+1. Rtn(n+1) is a set of new routes observed at time n+1 that were not present at time n.

RTc and RTn are each composed of two parts: one due to changes in network state (new routes appearing, changes to existing routes, etc.), and a second attributable to routing protocol changes (BGP session failure, BGP route attribute changes, changes to filtering policies, etc.). Equation (1) can be expanded to account for these separate effects. First, substitute equation (2) into equation (1):

RT(n+1) = RTo(n) + RTc(n+1) + RTn(n+1) (3)

As was mentioned, the terms RTc(n+1) and RTn(n+1) can be further expanded into their two constitute components:

RTc(n+1) = RTcN(n+1) + RTcR(n+1) (4)

RTn(n+1) = RTnN(n+1) + RTnR(n+1) (5)

In these two equations, "N" denotes the component due to network topology changes, and "R" denotes the component due to routing protocol changes.

These equations can be used as the basis for deriving the convergence and stability metrics discussed in Section 4. However, there are a number of issues that will need to be resolved in order to make progress:

a) Some thought will need to be done on how to distinguish between network and routing protocol effects;

b) Some thought needs to be given to "timescales of applicability" in order to make assessments about what constitutes instability in a routing system from a practical point-of-view;

c) Some thought needs to given to how a protocol can absorb network instabilities. [RFC2902] touches on this issue and indicated that damping the effects of route updates enhances stability, but possibly at the cost of reachability for some prefixes.

5.2 Derivation of stability metrics

In this section we propose an algorithm for calculating a stability metric for a route and a routing table.

First, we should make an attempt to quantify what we mean by stable, marginally stable, and unstable in the context of the routing table RT(n+1). Please note that this work is preliminary and is still in the process of being refined and tested.

We can start with the basic equation we previously developed:

RT(n+1) = RTo(n) + deltaRT(n+1)

Let |deltaRT(n+1)| be the magnitude of the change to the routing table at some time n+1.

For a routing table, RT(n+1), to be stable, the following condition must be met:

|deltaRT(n+1)| =< alpha as t -> infinity,

where alpha is a small, positive number.

For marginally stable systems, the following condition must be met:

alpha < |deltaRT(n+1)| =< beta as t -> infinity,

where beta is a small, positive number, greater than alpha.

For unstable systems, the following condition is met:

|deltaRT(n+1)| > beta as t -> infinity.

One can see that I have not made distinctions for new routes or changed routes, or for the source of disturbances to the system. This is a definition of stability at the highest, or coarsest, level.

As well, alpha and beta are going to need to be set based on some sort of operational criteria. Among other things, alpha and beta will be dependent on the observation sampling frequency.

In order to be able to compute |deltaRT(n+1)| we need to be able to calculate a stability metric for an individual route.

A route, rti(n+1), which is a component of RT(n+1), consists of:

rti(n+1) = {destination, path, attributes}.

A stability metric for rti might be most easily defined by an algorithm and in the next several paragraphs we will undertake such a development.

Let the stability metric associated with a route rti be called fi. When a route is created, the initial value of fi is 0.

If rti never experiences any change, then fi remains constant at 0.

If rti does experience a change (path or attribute or withdrawal), then fi changes according to the following:

```
if rti(n+1) != rti(n) then

   /* the route has changed */

   fi(n+1) = fi(n) + 1

else

   /* the route did not change */

   if fi(n) = 0 then
```

```
           /* fi never drops to less than 0 */
```

```
         fi(n+1) = 0

      else

        /* fi is decremented if there is no change in rti */

         fi(n+1) = f(n) - 1

      end if

    end if
```

So, how does this work in the case where rti is withdrawn at some time n+1? Conceivably, fi(n+1) is 1 at a minimum when withdrawal occurs, and some non-zero value fi(n)+1, say gamma, at most according to the algorithm. As t increases, fi is kept around until it equals zero, at which time the route, rti, is discarded.

With this definition of a stability metric for an individual route, one can take a stab at calculating a stability metric for an entire routing table.

|deltarti(n+1)| is introduced as the change in stability metric of a single route, rti, from t=n to t=n+1. It is used to calculate |deltaRT(n+1)|, the stability metric of the entire routing table, RT, at time t=n+1.

|deltaRT(n+1)| is normalized so that 0 is the minimum value and 1 is the maximum, where 0 implies perfect stability, and so that 1 indicates complete instability.

Here is the candidate algorithm to evaluate |deltaRT(n+1)|:

```
  for i = 1 to number of routes in RT(n+1)

     if rti(n+1) is a new route then

        |deltarti(n+1)| = 0

     else

        /* rti(n+1) is an existing route */

        if fi(n) = 0 and fi(n+1) = 0 then

           /* no change occured to the route */

           |deltarti(n+1)| = 0

        else
```

```
/* a change occured to the route */

if fi(n+1) > fi(n) then
```

```
                 |deltarti(n+1)| = fi(n+1) / [fi(n+1) + 1]

              else

                 |deltarti(n+1)| = fi(n+1) / fi(n)

              end if

           end if

        end if

     end i loop
```

|deltaRT(n+1)| = Sum(deltarti(n+1)) / total number of routes in RT(n+1)

We will conclude this section by illustrating some notable properties and showing some example calculations.

It should be noted that:

- fi(n+1) and fi(n) can only be equal if they are both equal to 0 otherwise, fi(n+1) and fi(n) only differ by 1, and there is no theoretical upper limit on either fi(n+1) or fi(n).
- 0 =< |deltarti(n+1)| =< 1, but, |deltarti(n+1)| only takes on values from the set: {0, 0/1, 1/2, 2/3, 3/4, 4/5, .. , m/m+1, ..}
- 0 =< |deltaRT(n+1)| =< 1, but, the particular values it assumes are continously variable from 0 to 1 unlike |deltarti(n+1)|.

The following 3 example calculations show values for |deltaRT(n+1)| in a number of simple, but indicative situations.

Example 1:

fi(n) = {0, 1, 2, 1, 0, 0} and fi(n+1) = {1, 2, 1, 0, 0, 0}

|deltaRT(n+1)| = (1/2 + 2/3 + 1/2 + 0/1 + 0 + 0) / 6
|deltaRT(n+1)| = 0.278 (rather stable)

Example 2:

fi(n) = {0, 0, 0, 0, 0, 0} and fi(n+1) = {1, 1, 1, 1, 1, 1}

|deltaRT(n+1)| = (0/1 + 0/1 + 0/1 + 0/1 + 0/1 + 0/1) / 6
|deltaRT(n+1)| = 0 (still stable, too early to judge)

Example 3:

fi(n) = {56, 20, 63, 64, 0, 5} and fi(n+1) = {57, 19, 64, 65, 0, 4}

```
|deltaRT(n+1)| = (57/58 + 19/20 + 64/65 + 65/66 + 0 + 4/5) / 6
|deltaRT(n+1)| = 0.784 (very unstable)
```

6. Previous work on BGP and Routing system stability

There have been numerous studies of BGP dynamics over the years. In subsequent versions of this draft, they will be summarized in this section and general findings will be drawn.

In this version of the document, we will just outline some of the findings surrounding recent studies concerned with interactions of BGP with Route Flap Damping (RFD) in order to show some of the complexity in understanding BGP dynamics.

Work began in the early 1990s on an enhancement to the BGP called "Route Flap Damping" [RFC2439]. The purpose of RFD was to prevent or limit sustained route oscillations that could potentially put an undue processing load on BGP. At that time there was a belief that the predominate cause of route oscillation was due to BGP routing sessions going up and down because they were being carried on circuits that were themselves persistently going up and down (see [Huston07b] for a fuller discussion). This would result in a constant stream of route updates and withdrawals from the affected BGP sessions that could propagate through the entire network due to the network's flat addressing architecture. The first draft of the RFD algorithm specification appeared in October 1993, updates and revisions lead to the publication of RFC 2439, BGP Route Flap Damping, in November 1998 [RFC2439].

Over the next several years, RIPE published three recommendations, [RIPE178], [RIPE210] and [RIPE229] in an attempt to establish guidelines for operators when setting RFD's user configurable parameters. The ultimate goal was to make the deployment of RFD consistent throughout the network because different vendors provided different default values for RFD's various parameters, and this could result in different damping behaviors across the network. The last of these recommendations, [RIPE229], was published in October 2001.

In August 2002, Mao et al. [Mao02] published a paper that discussed how the use of RFD, as specified in RFC 2439. They showed that RFD can significantly slowdown the convergence times of relatively stable routing entries. This abnormal behavior arises during route withdrawal from the interaction of RFD with "BGP path exploration" (in which in response to path failures or routing policy changes, some BGP routers may try a sequence of transient alternate paths before selecting a new path or declaring destination unreachability). The NANOG 2002 presentation of Bush et al. [Bush02] succinctly summarized the findings of Mao et al. [Mao02] and presented some observational data to illustrate the phenomena. The overall conclusion of this work was that it was best not to use RFD so that the overall ability of the network to re-converge after an episode of "BGP path exploration" was not needlessly slowed. In May 2006, RIPE published a final set of RFD recommendations [RIPE378] that directed operators to not use RFD due primarily to the findings presented in [Mao02].

Recently, solutions such as EPIC [Chandrashekar05], or improving BGP convergence through Root Cause Notification (BGP-RCN) [Pei05] have been proposed to solve the "BGP path exploration" problem; however, there are several details that still require consideration.

BGP stability has also been reported in [RFC4984], outcome of the Routing and Addressing Workshop held by the Internet Architecture Board (IAB).

7. Security Considerations

TBD.

8. IANA Considerations

This document makes no requests to IANA for action.

9. References

9.1 Normative References

[RFC2902] S.Deering, et al., "Overview of the 1998 IAB Routing Workshop", RFC 2902, August 2000.

[RFC2439] Villamizar, C., Chandra, R., and Govindan, R., "BGP Route Flap Damping", RFC 2439, November 1998.

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

[RFC4984] D.Meyer, Ed., L.Zhang, Ed., K.Fall, Ed., "Report from the IAB Workshop on Routing and Addressing", RFC 4984, September 2007.

9.2 Informative References

[Bush02] Bush, R., Griffin, T., and Mao, Z.M., "Route flap damping harmful?", NANOG-26, 28 October 2002. http://www.nanog.org/mtg-0210/ppt/flap.pdf

[Chandrashekar05] J.Chandrashekar, Z.Duan, Z.-L.Zhang, and J.Krasky, Limiting path exploration in BGP, In Proc. IEEE INFOCOM 2005, Miami, Florida, March 2005.

[Huston07a] G.Huston, http://bgp.potaroo.net, 2007.

[Huston07b] G.Huston, "Damping BGP", June 2007, http://www.potaroo.net/ispcol/2007-06/dampbgp.html

[Labovitz00] C.Labovitz, A.Ahuja, A.Bose, and F.Jahanian, "Delayed

Internet Routing Convergence," in Proceedings of ACM
SIGCOMM'00.

[Li07] T.Li, G.Huston, "BGP Stability Improvements", Internet draft, work in progress, draft-li-bgp-stability-01, June 2007.

[Mao02] Z.Mao, R.Govindan, G.Varghese, and R.Katz, "Route Flap Damping Exacerbates Internet Routing Convergence", ACM SIGCOMM'02, August 2002.

[Pei05] D.Pei, M.Azuma, D.Massey, and L.Zhang, "BGP-RCN: improving BGP convergence through root cause notification", Computer Networks, ISDN Syst. vol. 48, no. 2, pp 175-194, June 2005.

[RIPE178] Barber, T., Doran, S., Karrenberg, D., Panigl, C., and Schmitz, J., "RIPE Routing-WG Recommendations for coordinated route-flap damping parameters", RIPE-178, 2 February 1998. http://www.ripn.net:8080/nic/ripe-docs/ripe-178.txt

[RIPE210] Barber, T., Doran S., Karrenberg, Pangil, C., and Schmitz, J., "RIPE Routing-WG Recommendation for coordinated route-flap damping parameters", RIPE-210, 12 May 2000. http://www.ripn.net/nic/ripe-docs/ripe-210.txt

[RIPE229] Panigl, C., Schmitz, J., Smith, P., and Vistoli, C., "RIPE Routing-WG Recommendations for Coordinated Route-flap Damping Parameters", RIPE-229, 22 October 2001. ftp://ftp.ripe.net/ripe/docs/ripe-229.txt

[RIPE378] Smith, P., and Panigl, C., "RIPE Routing Working Group Recommendations on Route-flap Damping", RIPE-378, 11 May 2006. http://www.ripe.net/ripe/docs/ripe-378.html

10. Authors' Addresses

Dimitri Papadimitriou
Alcatel-Lucent
Copernicuslaan 50
B-2018 Antwerpen, Belgium
Phone: +32 3 2408491
Email: dimitri.papadimitriou@alcatel-lucent.be

James Lowe
Alcatel-Lucent
600 March Road
Ottawa, Ontario
Canada, K2K 2E6
Phone: 1-613-784-1495
Email: jim.lowe@alcatel-lucent.com

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).