

Benchmarking Methodology
Internet-Draft
Intended status: Informational
Expires: February 16, 2012

F. Baker
Cisco Systems
August 15, 2011

Testing Eyeball Happiness
draft-baker-bmwg-testing-eyeball-happiness-05

Abstract

The amount of time it takes to establish a session using common transport APIs in dual stack networks and networks with filtering such as proposed in BCP 38 is a barrier to IPv6 deployment. This note describes a test that can be used to determine whether an application can reliably establish sessions quickly in a complex environment such as dual stack (IPv4+IPv6) deployment or IPv6 deployment with multiple prefixes and upstream ingress filtering. This test is not a test of a specific algorithm, but of the external behavior of the system as a black box. Any algorithm that has the intended external behavior will be accepted by it.

Requirements

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].

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 <http://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 February 16, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Measuring Eyeball Happiness	4
2.1. Happy Eyeballs test bed configuration	4
2.2. Happy Eyeballs test procedure	6
2.3. Happy Eyeballs metrics	7
2.3.1. Metric: Session Setup Interval	7
2.3.2. Metric: Maximum Session Setup Interval	8
2.3.3. Metric: Minimum Session Setup Interval	9
2.3.4. Descriptive Metric: Attempt pattern	9
3. IANA Considerations	10
4. Security Considerations	10
5. Acknowledgements	10
6. Change Log	10
7. References	10
7.1. Normative References	10
7.2. Informative References	11
Author's Address	11

1. Introduction

The Happy Eyeballs [I-D.ietf-v6ops-happy-eyeballs] specification notes an issue in deployed multi-prefix IPv6-only and dual stack networks, and proposes a correction. [RFC5461] similarly looks at TCP's response to so-called "soft errors" (ICMP host and network unreachable messages), pointing out an issue and a set of possible solutions.

In a dual stack network (i.e., one that contains both IPv4 [RFC0791] and IPv6 [RFC2460] prefixes and routes), or in an IPv6-only network that uses multiple prefixes allocated by upstream providers that implement BCP 38 Ingress Filtering [RFC2827], the fact that two hosts that need to communicate have addresses using the same architecture does not imply that the network has usable routes connecting them, or that those addresses are useful to the applications in question. In addition, the process of establishing a session using the Sockets API [RFC3493] is generally described in terms of obtaining a list of possible addresses for a peer (which will normally include both IPv4 and IPv6 addresses) using `getaddrinfo()` and trying them in sequence until one succeeds or all have failed. This naive algorithm, if implemented as described, has the side-effect of making the worst case delay in establishing a session far longer than human patience normally allows.

This has the effect of discouraging users from enabling IPv6 in their equipment, or content providers from offering AAAA records for their services.

This note describes a test to determine how quickly an application can reliably open sessions in a complex environment, such as dual stack (IPv4+IPv6) deployment or IPv6 deployment with multiple prefixes and upstream ingress filtering. This is not a test of a specific algorithm, but a measurement of the external behavior of the application and its host system as a black box. The "happy eyeballs" question is this: how long does it take an application to open a session with a server or peer, under best case and worst case conditions?

The methods defined here make the assumption that the initial communication set-up of many applications can be summarized by the measuring the DNS query/response and transport layer handshaking, because no application-layer communication takes place without these steps.

The methods and metrics defined in this note are ideally suited for Laboratory operation, as this affords the greatest degree of control to modify configurations quickly and produce consistent results.

However, if the device under test is operated as a single user with limited query and stream generation, then there's no concern about overloading production network devices with a single "set of eyeballs". Therefore, these procedures and metrics MAY be applicable to production network application, as long as the injected traffic represents a single user's typical traffic load, and the testers adhere to the precautions of the relevant network with respect to re-configuration of devices in production.

2. Measuring Eyeball Happiness

This measurement determines the amount of time it takes an application to establish a session with a peer in the presence of at least one IPv4 and multiple IPv6 prefixes and a variety of network behaviors. ISPs are reporting that a host (MacOSX, Windows, Linux, FreeBSD, etc) that has more than one address (an IPv4 and an IPv6 address, two global IPv6 addresses, etc) may serially try addresses, allowing each TCP setup to expire, taking several seconds for each attempt. There have been reports of lengthy session setup times - in various application and OS combinations anywhere from multi-second to half an hour - as a result. The amount of time necessary to establish communication between two entities should be approximately the same regardless of the type of address chosen or the viability of routing in the specific network; users will expect this time to be consistent with their current experience (else, happiness is at risk).

2.1. Happy Eyeballs test bed configuration

The configuration of equipment and applications is as shown in Figure 1.

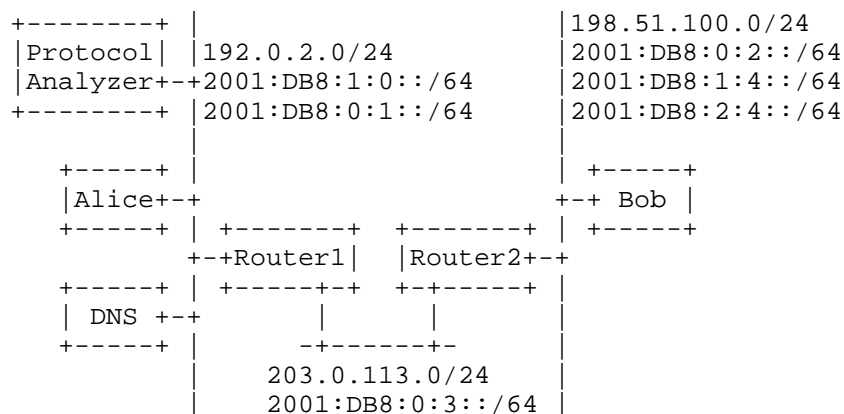


Figure 1: Generic Test Environment

Alice is the unit being measured, the computer running the process that will establish a session with Bob for the application in question. DNS is represented in the diagram as a separate system, as is the protocol analyzer that will watch Alice's traffic. This is not absolutely necessary; If one computer can run tcpdump and a DNS server process - and for that matter subsume the routers - that is acceptable. The units are separated in the test for purposes of clarity.

On each test run, configuration is performed in Router 1 to permit only one route to work. There are various ways this can be accomplished, including but not limited to installing

- o a filter that drops datagrams to Bob resulting in an ICMP "administratively prohibited",
- o a filter that silently drops datagrams to Bob,
- o a null route or removing the route to one of Bob's prefixes, resulting in an ICMP "destination unreachable", and
- o a middleware program that responds with a TCP RST.
- o Path MTU issues

The Path MTU Discovery [RFC1191][RFC1981] matter requires some explanation. With IPv6, and with IPv4 when "Do Not Fragment" is set, a router with a message too large for an interface discards it and replies with an ICMPv4 "Destination Unreachable: Datagram Too Big" or ICMPv6 "Packet Too Big". If this packet is lost, the source doesn't know what size to fragment to and has no indication that fragmentation is required. A configuration for this scenario would set the MTU on 203.0.113.0/24 or 2001:DB8:0:3::/64 to the smallest allowed by the address family (576 or 1280) and disable generation of the indicated ICMP message. Note that [RFC4821] is intended to address these issues.

The tester should try different methods to determine whether differences in this configuration make a difference in the test. For example, one might find that the application under test responds differently to a TCP RST than to a silent packet loss. Each of these scenarios should be tested; if doing so is too difficult, the most important is the silent packet loss case, as it is the worst case.

2.2. Happy Eyeballs test procedure

Consider a network as described in Section 2.1. Alice and Bob each have a set of one or more IPv4 and two or more IPv6 addresses. Bob's are in DNS, where Alice can find them; Alice's and others may be there as well, but are not relevant to the test. Routers 1 and 2 are configured to route the relevant prefixes. Different measurement trials revise an access list or null route in Router 1 that would prevent traffic Alice->Bob using each of Bob's addresses. If Bob has a total of N addresses, we run the measurement at least N times, permitting exactly one of the addresses to enjoy end to end communication each time. If the DNS service randomizes the order of the addresses, this may not result in a test requiring establishment of a connection to all of the addresses; in this case, the test will have to be run repeatedly until in at least one instance a TCP SYN or its equivalent is seen for each relevant address. The tester should either flush the resolver cache between iterations, to force repeated DNS resolution, or should wait for at least the DNS RR TTL on each resource record. In the latter case, the tester should also observe DNS re-resolving; if not, the application is not correctly using DNS.

This specification assumes common LAN technology with no competing traffic and nominal propagation delays, so that they are not a factor in the measurement.

The objective is to measure the amount of time required to establish a session. This includes the time from Alice's initial DNS request through one or more attempts to establish a session to the session being established, as seen in the LAN trace. The simplest way to measure this will be to put a traffic analyzer on Alice's point of attachment and capture the messages exchanged by Alice.

DNS Server	Alice	Bob
1.	<--www.example.com A-----	
2.	<--www.example.com AAAA---	
3.	---198.51.100.1----->	
4.	---2001:DB8:0:2::1----->	
5.		
6.	--TCP SYN, IPv6--->X	<*****
7.	--TCP SYN, IPv6--->X	
8.	--TCP SYN, IPv6--->X	TCP 3wHS
9.		Time
10.	--TCP SYN, IPv4----->	(any family)
11.	<-TCP SYN+ACK, IPv4----	
12.	--TCP ACK, IPv4----->	<*****

Figure 2: Message flow using TCP

In a TCP-based application (Figure 2), that would be from the DNS request on line 1 through the first completion of a TCP three-way handshake, the transmission on line 12.

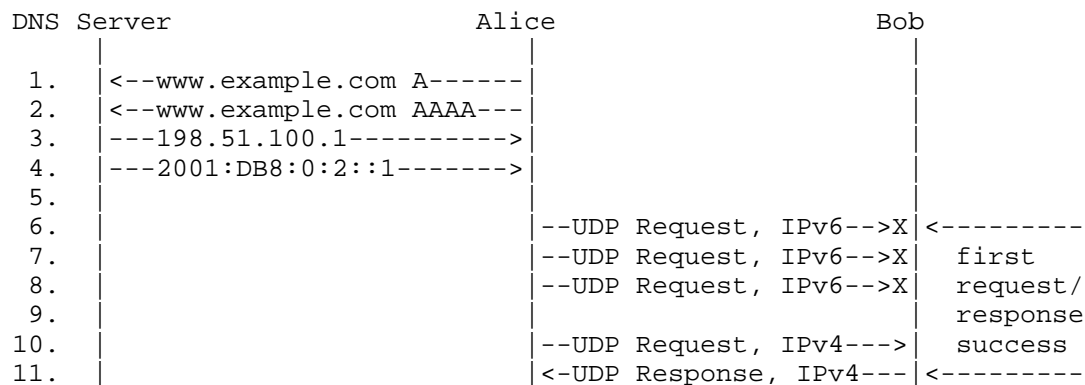


Figure 3: Message flow using UDP

In a UDP-based application (Figure 3), that would be from the DNS request (line 1) through one or more UDP Requests (lines 6-10) until a UDP Response is seen (line 11).

When using other transports, the methodology will have to be specified in context; it should measure the same event.

2.3. Happy Eyeballs metrics

The measurements taken are the duration of the interval from the initial DNS request until the session is seen to have been established, as described in Section 2.2. We are interested in the shortest and longest durations (which will most likely be those that send one SYN and succeed and those that send a SYN to each possible address before succeeding in one of the attempts), and the pattern of attempts sent to different addresses. The pattern may be to simply send an attempt every <time interval>, or may be more complex; as a result, this is in part descriptive.

ALL measurement events on the sending and receiving of messages SHALL be observed at the "Alice" attachment point and time stamps SHOULD be applied upon reception of the last bit of the IP information field. Use of a alternate timing reference SHALL be noted.

2.3.1. Metric: Session Setup Interval

Name: Session Setup Interval

Description: The session setup interval MUST be the time beginning with the first DNS query sent (observed at Alice's attachment), and ending with successful transport connection establishment (as indicated in line 12 of Figure 2, and line 11 of Figure 3). This interval is defined as the session setup interval.

This test will be run several times, once for each possible combination of destination address (configured on Bob) and failure mode (configured on Router 1).

Methodology: In the LAN analyzer trace, note the times of the initial DNS request and the confirmation that the session is open as described in Section 2.2. If the session is not successfully opened, possibly due to Alice aborting the attempt, the Session Setup Interval is considered to be infinite.

Units: Session setup time is measured in milliseconds.

Measurement Point(s): The measurement point is at Alice's LAN interface, both sending and receiving, observed using a program such as tcpdump running on Alice or an external analyzer.

Timing: The measurement program or external analyzer MUST run for a duration sufficient to capture the entire message flow as described in Section 2.2. Measurement precision MUST be sufficient to maintain no more than 0.1 ms error over a 60 second interval. 1 ppm precision would suffice.

2.3.2. Metric: Maximum Session Setup Interval

Name: Maximum Session Setup Interval

Description: The maximum session setup interval is the longest period of time observed for the establishment of a session as described in Section 2.3.1.

Methodology: see Session Setup Interval.

Units: Session setup time is measured in milliseconds.

Measurement Point(s): see Session Setup Interval.

Timing: The measurement program or external analyzer MUST run for a duration sufficient to capture the entire message flow as described in Section 2.2. Measurement precision MUST be sufficient to maintain no more than 0.1 ms error over a 60 second

interval. 1 ppm precision would suffice.

2.3.3. Metric: Minimum Session Setup Interval

Name: Minimum Session Setup Interval

Description: The minimum session setup interval is the shortest period of time observed for the establishment of a session.

Methodology: see Session Setup Interval.

Units: Session setup time is measured in milliseconds.

Measurement Point(s): see Session Setup Interval.

Timing: The measurement program or external analyzer MUST run for a duration sufficient to capture the entire message flow as described in Section 2.2. Measurement precision MUST be sufficient to maintain no more than 0.1 ms error over a 60 second interval. 1 ppm precision would suffice.

2.3.4. Descriptive Metric: Attempt pattern

Name: Attempt pattern

Description: The Attempt Pattern is a description of the observed pattern of attempts to establish the session. In simple cases, it may be something like "Initial TCP SYNs to a new address were observed every <so many> milliseconds"; in more complex cases, it might be something like "Initial TCP SYNs in IPv6 were observed every <so many> milliseconds, and other TCP SYNs using IPv4 were observed every <so many> milliseconds, but the two sequences were independent." It may also comment on retransmission patterns if observed.

Methodology: The traffic trace is analyzed to determine the pattern of initiation.

Units: milliseconds.

Measurement Point(s): The measurement point is at Alice's LAN interface, observed using a program such as tcpdump running on Alice or an external analyzer.

Timing: The measurement program or external analyzer MUST run for a duration sufficient to capture the entire message flow as described in Section 2.2. Measurement precision MUST be sufficient to maintain no more than 0.1 ms error over a 60 second

interval. 1 ppm precision would suffice.

3. IANA Considerations

This memo asks the IANA for no new parameters.

Note to RFC Editor: This section will have served its purpose if it correctly tells IANA that no new assignments or registries are required, or if those assignments or registries are created during the RFC publication process. From the author's perspective, it may therefore be removed upon publication as an RFC at the RFC Editor's discretion.

4. Security Considerations

This note doesn't address security-related issues.

5. Acknowledgements

This note was discussed with Dan Wing, Andrew Yourtchenko, and Fernando Gont. In the Benchmark Methodology Working Group, Al Morton, David Newman, Sarah Banks, and Tore Anderson made comments.

6. Change Log

-00: Initial version - November, 2010

-01: Rewritten per suggestions by Al Morton, David Newman, and Sarah Banks.

-02: Clean-up per working group comments.

-03: Updated per Al Morton's and Tore Anderson's comments.

7. References

7.1. Normative References

[I-D.ietf-v6ops-happy-eyeballs]

Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", draft-ietf-v6ops-happy-eyeballs-03 (work in progress), July 2011.

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

7.2. Informative References

- [RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.
- [RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, November 1990.
- [RFC1981] McCann, J., Deering, S., and J. Mogul, "Path MTU Discovery for IP version 6", RFC 1981, August 1996.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.
- [RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.
- [RFC3493] Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, February 2003.
- [RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, March 2007.
- [RFC5461] Gont, F., "TCP's Reaction to Soft Errors", RFC 5461, February 2009.

Author's Address

Fred Baker
Cisco Systems
Santa Barbara, California 93117
USA

Email: fred@cisco.com

INTERNET-DRAFT
Intended Status: Informational
Expires: April 1, 2012

T. Alexander, K. Green
(Ixia)
October 2011

Benchmarking Methodology for Evaluating the Security Effectiveness
of Content Aware Devices
draft-green-bmwg-seceff-bench-meth-00

Abstract

This document defines a methodology for evaluating the ability of content-aware network devices to correctly detect and block malicious or administratively disallowed traffic flows. This benchmark addresses the issue of classification accuracy under well defined conditions. It is not concerned with measuring forwarding performance which is covered by other BMWG documents.

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 <http://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 April 1, 2012.

Copyright and License Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1	Introduction	3
1.1	Requirements Language	3
2	Terminology	4
2.1	Existing Terminology	4
2.2	New Terminology	4
2.2.1	Good Traffic	4
2.2.2	Evil Traffic	4
3	Test Setup	4
3.1	Application Traffic Mix	4
4	Benchmarking Tests	5
4.1	Maximum Attack Blocking Rate	5
4.2	Useful Attack Blocking Rate	5
4.3	Attack Blocking Effectiveness	5
5	Security Considerations	5
6	IANA Considerations	6
7	Acknowledgements	6
8	References	6
8.1	Normative References	6
8.2	Informative References	6
	Authors' Addresses	6

1 Introduction

Networks of the 21st century exist in an environment flooded with complex and highly sophisticated security threats. There is an intense and enduring arms race under way between those developing and distributing attack technology and those developing and supplying defense technology.

In addition there is a growing need to limit access from users inside private or corporate networks to Internet sites or services deemed undesirable and to ensure that intellectual property and other private information is not allowed to pass freely from inside the protected network to the outside world.

In response to the this dynamic and constantly expanding range of security threats and privacy requirements there is a growing diversity of network devices that provide a variety of defensive services including but not limited to firewall, intrusion detection, intrusion prevention, anti-virus, anti-malware, anti-spam, anti-dos, anti-ddos, unified threat management, data leakage prevention and more. These content-aware devices use a mixture of stateless and stateful L3 to L7 technologies, including deep packet inspection (DPI) to categorize traffic flows.

What all of these defensive solutions have in common is the requirement that they reliably and accurately distinguish between evil (malicious or disallowed) traffic and good traffic.

Categorization of traffic as either good or evil is fundamental to the operation of these devices since it is a prerequisite to all security functions.

Security Effectiveness is a measure of how accurately the device under test (DUT) categorizes traffic:

- o No false negatives = correctly blocks all evil traffic
- o No false positives = never blocks good traffic

In contrast, Security Performance is the characterization of the DUT's forwarding performance while under attack. Security Performance measures how well the device forwards good traffic with security features enabled and in the presence of evil traffic. This is addressed in [HAMILTON].

Security Effectiveness is orthogonal to Security Performance.

1.1 Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2 Terminology

2.1 Existing Terminology

2.2 New Terminology

2.2.1 Good Traffic

Good traffic is any traffic flow which is benign and should not be blocked by the DUT.

2.2.2 Evil Traffic

Evil traffic is any illicit traffic flow which should be blocked by the DUT. Evil traffic is either malicious (i.e. part of a deliberate attack) or administratively banned (i.e. disallowed from passing in to or out of the protected network due to its content and/or destination).

3 Test Setup

3.1 Application Traffic Mix

Some test cases require the test equipment to inject good traffic mixed with the evil traffic. The purpose of the good traffic is to force the DUT to distinguish between good and evil traffic and it is not used to quantify the forwarding performance of the DUT from an application perspective.

Given this purpose, in order to protect the integrity and repeatability of the benchmark, a single fixed definition of the good traffic application mix is provided. No attempt is made to accurately model any particular mix of application traffic such as might be seen in an operational network.

Rather, the traffic mix includes an appropriate mix of traffic types to ensure that the security engine cannot blindly assume that every packet is either good or evil and so deliver unrealistically high performance or otherwise undermine the benchmark.

In those test scenarios where application traffic is specified, the

following mix MUST be used:

*** TBD but likely to include at least UDP, TCP, HTTP ***

4 Benchmarking Tests

4.1 Maximum Attack Blocking Rate

Maximum Attack Blocking Rate (attacks/second) is defined as the largest number of attacks per second where 100% of attacks are blocked with no application traffic present.

4.2 Useful Attack Blocking Rate

Useful Attack Blocking Rate (attacks/second) is defined as the largest number of attacks per second where 100% of attacks are blocked in the presence of good traffic and 0% of the good traffic is blocked or dropped.

4.3 Attack Blocking Effectiveness

Attack Blocking Effectiveness (percentage) is the ratio of blocked attacks/attempted attacks counted over the total number of different types of attack in the presence of good traffic and where 0% of the good traffic is dropped or blocked.

5 Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the other constraints defined in [RFC2544].

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT/SUT.

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

6 IANA Considerations

This memo includes no request to IANA.

7 Acknowledgements

Thanks to X, Y & Z for their review and comments.

8 References

8.1 Normative References

- [RFC1242] Bradner, S., "Benchmarking Terminology for Network Interconnection Devices", RFC 1242, July 1991.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

8.2 Informative References

- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [RFC3511] Hickman, B., Newman, D., Tadjudin, S., and T. Martin, "Benchmarking Methodology for Firewall Performance", RFC 3511, April 2003.
- [HAMILTON] "Benchmarking Methodology for Content Aware Network Devices", draft-hamilton-bmwg-ca-bench-07.txt

Authors' Addresses

Kenneth Green
Ixia
Australia

EMail: kgreen@ixiacom.com

Tom Alexander
Ixia
USA

EMail: talexander@ixiacom.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: September 8, 2011

M. Hamilton
BreakingPoint Systems
S. Banks
Cisco Systems
March 7, 2011

Benchmarking Terminology for Content-Aware Network Devices
draft-hamilton-bmwg-ca-bench-term-00

Abstract

The purpose of this document is to define and outline the terminology necessary to appropriately follow and implement "Benchmarking Methodology for Content-Aware Network Devices". Relevant terms will be defined and discussed throughout this document in order to ensure the comprehension of the previously mentioned methodology.

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 <http://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 September 8, 2011.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as

described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	4
2. Scope	4
3. Definitions	4
3.1. Application Flow	5
3.2. Application Throughput	5
3.3. Average Time to TCP Session Establishment	6
3.4. Content-Aware Device	6
3.5. Deep Packet Inspection	7
3.6. Network 5-Tuple	7
3.7. Session Establishment Rate	8
3.8. Session Establishment Time	8
3.9. Simultaneous TCP Sessions	9
3.10. Time To SYN	9
4. IANA Considerations	10
5. Security Considerations	10
6. References	10
6.1. Normative References	10
6.2. Informative References	11
Authors' Addresses	11

1. Introduction

Content-aware and deep packet inspection (DPI) device penetration has grown significantly over the last decade. No longer are devices simply using Ethernet headers and IP headers to make forwarding decisions. Devices that could historically be classified as 'stateless' or raw forwarding devices are now seeing more DPI functionality. Devices such as core and edge routers are now being developed with DPI functionality to make more intelligent routing and forwarding decisions.

The Benchmarking Working Group (BMWG) has historically produced Internet Drafts and Requests for Comment that are focused specifically on creating output metrics that are derived from a very specific and well-defined set of input parameters that are completely and unequivocally reproducible from testbed to testbed. The end goal of such methodologies is to, in the words of the BMWG charter "reduce specmanship" from network equipment manufacturers (NEM's). Existing BMWG work has certainly met this stated goal.

Today, device sophistication has expanded beyond existing methodologies, allowing vendors to reengage in specmanship. In order to achieve the stated BMWG goals, the methodologies designed to hold vendors accountable must evolve with the enhanced device functionality.

The BMWG has historically avoided the use of the term "realistic" throughout all of its drafts and RFCs. While this document will not explicitly use this term, the end goal of the terminology and methodology is to generate performance metrics that will be as close as possible to equivalent metrics in a production environment. It should be further noted that any metrics acquired from a production network MUST be captured according to the policies and procedures of the IPPM or PMOL working groups.

An explicit non-goal of this document is to replace existing methodology/terminology pairs such as RFC 2544 [1]/RFC 1242 [2] or RFC 3511 [3]/RFC 2647 [4]. The explicit goal of this document is to create a methodology and terminology pair that is more suited for modern devices while complementing the data acquired using existing BMWG methodologies. Existing BMWG work generally revolves around completely repeatable input stimulus, expecting fully repeatable output. This document departs from this mantra due to the nature of modern traffic and is more focused on output repeatability than on static input stimulus.

Some of the terms used throughout this draft have previously been defined in "Benchmarking Terminology for Firewall Performance" RFC

2647 [4]. This document SHOULD be consulted prior to using this document.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [5].

2. Scope

Content-aware devices take many forms, shapes and architectures. These devices are advanced network interconnect devices that inspect deep into the application payload of network data packets to do classification. They may be as simple as a firewall that uses application data inspection for rule set enforcement, or they may have advanced functionality such as performing protocol decoding and validation, anti-virus, anti-spam and even application exploit filtering.

This document is strictly focused on examining performance and robustness across a focused set of metrics that may be used to more accurately predict device performance when deployed in modern networks. These metrics will be implementation independent.

It should also be noted that the purpose of this document is not to perform functional testing of the potential features in the Device/System Under Test (DUT/SUT)[4] nor specify the configurations that should be tested. Various definitions of proper operation and configuration may be appropriate within different contexts. While the definition of these parameters are outside the scope of this document, the specific configuration of both the DUT and tester SHOULD be published with the test results for repeatability and comparison purposes.

While a list of devices that fall under this category will quickly become obsolete, an initial list of devices that would be well served by utilizing this type of methodology should prove useful. Devices such as firewalls, intrusion detection and prevention devices, application delivery controllers, deep packet inspection devices, and unified threat management systems generally fall into the content-aware category.

3. Definitions

3.1. Application Flow

Definition:

An application flow is the virtual connection between two network hosts that is used to exchange user data above the transport layer.

Discussion:

Content-aware devices may potentially proxy session-layer connections, acting as a virtual server to the client and a virtual client to the server. In this mode, the SUT/DUT may modify members of the network 5-tuple or act on their behalf, thus each end host is actually disconnected at the session layer. Application flows are virtual connections that are between the two hosts, irrespective of the nature of the session layer semantics.

Unit of Measurement:

N/A

Issues:

N/A

See Also:

5-Tuple

3.2. Application Throughput

Definition:

The rate at which data associated with an application flow is transmitted through the SUT/DUT.

Discussion:

Throughput metrics may be calculated at various layers in the network protocol stack. Each layer does contain associated overhead necessary to maintain that layer. Application throughput is the number of bits transmitted through a SUT/DUT, not including the overhead associated with lower layer protocols. Measurement should be taken at the receiver side to minimize the impact of session layer retransmissions.

Unit of Measurement:

N/A

Issues:

Some applications may not rely on session layer reliability mechanisms. This definition does not cover the case where an application may utilize its own specific reliability/retransmission algorithm.

See Also:
N/A

3.3. Average Time to TCP Session Establishment

Definition:

The average time that a SUT/DUT requires to complete the TCP session establishment process.

Discussion:

The average time to TCP session establishment is calculated by taking the sum of all "TCP Session Establishment Time" values acquired in the specified time frame and divide by the total number of sessions established within that timeframe. The timeframe in which the average is taken will depend on the methodology itself and what is trying to be measured.

Unit of Measurement:
Seconds.

Issues:

Depending on how the DUT/SUT handles TCP session establishment, the client and server may have different values for the same TCP session. A client-side session may be established prior to the server-side session being established.

See Also:
See Also.

3.4. Content-Aware Device

Definition:

A networking device which performs deep packet inspection.

Discussion:

For a more detailed discussion, please see "deep packet inspection".

Unit of Measurement:
Not Applicable.

Issues:

Not Applicable.

See Also:
Deep Packet Inspection

3.5. Deep Packet Inspection

Definition:

The process by which a network device inspects layer 7 payload as well as protocol headers when making processing decisions.

Discussion:

Deep packet inspection (DPI) has grown from a feature reserved for Intrusion Prevention Devices into functionality that is shared across many next generation networking devices. Devices traditionally classified as firewalls are now looking at layer 7 payloads to make decisions, whether it is classification, rate-shaping, or actually deeming whether a flow is allowed. Many deep-packet inspection devices utilize proxy behavior as a functional choice for performing inspection.

Unit of Measurement:

Not Applicable.

Issues:

Not Applicable.

See Also:

Content-Aware Device

3.6. Network 5-Tuple

Definition:

The set of 5 metrics which distinguish two session layer connections from each other.

Discussion:

When discussing data transfer between hosts, a Network 5-tuple is typically used to differentiate between multiple session layer connections. Source and destination IP addresses, source and destination session-layer ports, and the session layer protocol make up the network 5-tuple. The session layer protocol is typically TCP or UDP, but may be SCTP or another session layer protocol.

Unit of Measurement:

N/A

Issues:

N/A

3.7. Session Establishment Rate

Definition:

The rate at which TCP sessions may be established through a given DUT/SUT.

Discussion:

The session establishment rate is a measurement of how many TCP sessions the DUT/SUT is able to establish in a given unit of time. If within a 1 second time interval the tester is able to establish 10,000 sessions, that rate will be measured at 10,000 sessions per second. The session must be established in accordance with the policy set forth in "Session Establishment Time".

Unit of Measurement:

TCP session(s) per second

Issues:

Issues.

See Also:

See Also.

3.8. Session Establishment Time

Definition:

Session establishment time is the difference in time between the first TCP SYN packet sent from the client and when TCP ACK packet's arrival at the server interface.

Discussion:

This metric is calculated between the time the first bit of the TCP SYN packet is sent from the client and the time the last bit of the TCP ACK packet arrives on the server interface.

Unit of Measurement:

Seconds.

Issues:

Depending on how the DUT/SUT handles TCP session establishment, the client and server may have different values for the same logical TCP session. A client-side session may be established prior to the server-side session being established.

See Also:

3.9. Simultaneous TCP Sessions

Definition:

The number of TCP sessions which are in the 'Established State' as defined by RFC 793 [6].

Discussion:

This measurement counts the number of TCP sessions which are in the 'Established State'. Sessions which are in this state must be able to maintain data transfer between client and server, bi-directionally.

Unit of Measurement:

Sessions.

Issues:

Depending on the nature of the SUT/DUT, the number of simultaneous sessions may instantaneously be different when counted from the client and server sides of the SUT/DUT.

See Also:

See Also.

3.10. Time To SYN

Definition:

The Time to SYN is a one-way metric, which is the difference between the time that the first TCP SYN packet is sent by the client and the time at which the server receives the TCP SYN packet from the client.

Discussion:

This metric is more important with content-aware devices due to the potential proxying issues. Content-aware devices may proxy a TCP session on behalf of the server. Many times, the client will receive the SYN/ACK from the DUT/SUT and complete the TCP handshake before the SYN has been forwarded to the server. This measurement is actually a proxy measure for client-side session establishment time through the DUT/SUT, if the session is in fact proxied.

Unit of Measurement:

Seconds.

See Also:

4. IANA Considerations

This memo includes no request to IANA.

All drafts are required to have an IANA considerations section (see the update of RFC 2434 [9] for a guide). If the draft does not require IANA to do anything, the section contains an explicit statement that this is the case (as above). If there are no requirements for IANA, the section will be removed during conversion into an RFC by the RFC Editor.

5. Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the other constraints RFC 2544 [1].

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network

6. References

6.1. Normative References

- [1] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [2] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [3] Hickman, B., Newman, D., Tadjudin, S., and T. Martin, "Benchmarking Methodology for Firewall Performance", RFC 3511, April 2003.
- [4] Newman, D., "Benchmarking Terminology for Firewall Performance", RFC 2647, August 1999.
- [5] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [6] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

- [7] Popoviciu, C., Hamza, A., Van de Velde, G., and D. Dugatkin, "IPv6 Benchmarking Methodology for Network Interconnect Devices", RFC 5180, May 2008.
- [8] Brownlee, N., Mills, C., and G. Ruth, "Traffic Flow Measurement: Architecture", RFC 2722, October 1999.

6.2. Informative References

- [9] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

Authors' Addresses

Mike Hamilton
BreakingPoint Systems
Austin, TX 78717
US

Phone: +1 512 636 2303
Email: mhamilton@breakingpoint.com

Sarah Banks
Cisco Systems
San Jose, CA 95134
US

Email: sabanks@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 22, 2012

S. Bradner
Harvard University
K. Dubray
Juniper Networks
J. McQuaid
Turnip Video
A. Morton
AT&T Labs
October 20, 2011

RFC 2544 Applicability Statement: Use on Production Networks Considered
Harmful
draft-ietf-bmwg-2544-as-01

Abstract

Benchmarking Methodology Working Group (BMWG) has been developing key performance metrics and laboratory test methods since 1990, and continues this work at present. Recent application of the methods beyond their intended scope is cause for concern. This memo clarifies the scope of RFC 2544 and other benchmarking work for the IETF community.

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 <http://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 April 22, 2012.

Copyright Notice

Copyright (c) 2011 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

(<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	3
2. Scope and Goals	4
3. The Concept of an Isolated Test Environment	4
4. Why RFC 2544 Methods are intended for ITE	4
4.1. Experimental Control, Repeatability, and Accuracy	4
4.2. Containment of Implementation Failure Impact	5
5. Advisory on RFC 2544 Methods in Real-world Networks	5
6. What to do without RFC 2544?	6
7. Security Considerations	6
8. IANA Considerations	7
9. Acknowledgements	7
10. References	7
10.1. Normative References	7
10.2. Informative References	8
Authors' Addresses	8

1. Introduction

This memo clarifies the scope of RFC 2544 [RFC2544], and other benchmarking work for the IETF community.

Benchmarking Methodologies (beginning with [RFC2544]) have always relied on test conditions that can only be reliably produced in the laboratory. Thus it was surprising to find that this foundation methodology was being cited in several unintended applications, such as:

1. Validation of telecommunication service configuration, such as the Committed Information Rate (CIR).
2. Validation of performance metrics in a telecommunication Service Level Agreement (SLA), such as frame loss and latency.
3. As an integral part of telecommunication service activation testing, where traffic that shares network resources with the test might be adversely affected.

Above, we distinguish "telecommunication service" (where a network service provider contracts with a customer to transfer information between specified interfaces at different geographic locations in the real world) from the generic term "service". Also, we use the adjective "production" to refer to networks carrying live user traffic. [RFC2544] used the term "real-world" to refer production networks and to differentiate them from test networks.

Although RFC 2544 is held up as the standard reference for such testing, we believe that the actual methods used vary from RFC 2544 in significant ways. Since the only citation is to RFC 2544, the modifications are opaque to the standards community and to users in general (an undesirable situation).

To directly address this situation, the past and present Chairs of the IETF Benchmarking Methodology Working Group (BMWG) have prepared this Applicability Statement for RFC 2544.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Scope and Goals

This memo clarifies the scope of [RFC2544], with the goal to provide guidance to the community on its applicability, which is limited to laboratory testing.

3. The Concept of an Isolated Test Environment

An Isolated Test Environment (ITE) used with [RFC2544] methods (as illustrated in Figures 1 through 3 of [RFC2544]) has the ability to:

- o contain the test streams to paths within the desired set-up
- o prevent non-test traffic from traversing the test set-up

These features allow unfettered experimentation, while at the same time protecting equipment management LANs and other production networks from the unwanted effects of the test traffic.

4. Why RFC 2544 Methods are intended for ITE

The following sections discuss some of the reasons why RFC 2544 [RFC2544] methods were intended only for isolated laboratory use, and the difficulties of applying these methods outside the lab environment.

4.1. Experimental Control, Repeatability, and Accuracy

All of the tests described in RFC 2544 assume that the tester and device under test are the only devices on the networks that are transmitting data. The presence of other unwanted traffic on the network would mean that the specified test conditions have not been achieved.

Assuming that the unwanted traffic appears in variable amounts over time, the repeatability of any test result will likely depend to some degree on the unwanted traffic.

The presence of unwanted or unknown traffic makes accurate, repeatable, and consistent measurements of the performance of the device under test very unlikely, since the actual test conditions will not be reported.

For example, the RFC 2544 Throughput Test attempts to characterize a maximum reliable load, thus there will be testing above the maximum that causes packet/frame loss. Any other sources of traffic on the

network will cause packet loss to occur at a tester data rate lower than the rate that would be achieved without the extra traffic.

4.2. Containment of Implementation Failure Impact

RFC 2544 methods, specifically to determine Throughput as defined in [RFC1242] and other benchmarks, may overload the resources of the device under test, and may cause failure modes in the device under test. Since failures can become the root cause of more wide-spread failure, it is clearly desirable to contain all DUT traffic within the ITE.

In addition, such testing can have a negative affect on any traffic which shares resources with the test stream(s) since, in most cases, the traffic load will be close to the capacity of the network links.

Appendix C.2.2 of [RFC2544] (as adjusted by errata) gives the private IPv4 address range for testing:

"...The network addresses 198.18.0.0 through 198.19.255.255 have been assigned to the BMWG by the IANA for this purpose. This assignment was made to minimize the chance of conflict in case a testing device were to be accidentally connected to part of the Internet. The specific use of the addresses is detailed below."

In other words, devices operating on the Internet may be configured to discard any traffic they observe in this address range, as it is intended for laboratory ITE use only. Thus, testers using the assigned testing address ranges MUST NOT be connected to the Internet.

We note that a range of IPv6 addresses has been assigned to BMWG for laboratory test purposes, in [RFC5180]. Also, the strong statements in the Security Considerations Section of this memo make the scope even more clear; this is now a standard fixture of all BMWG memos.

5. Advisory on RFC 2544 Methods in Real-world Networks

The tests in [RFC2544] were designed to measure the performance of network devices, not of networks, and certainly not production networks carrying user traffic on shared resources. There will be unanticipated difficulties when applying these methods outside the lab environment.

Operating test equipment on production networks according to the methods described in [RFC2544], where overload is a possible outcome, would no doubt be harmful to user traffic performance. These tests

MUST NOT be used on active networks. And as discussed above, the tests will never produce a reliable or accurate benchmarking result.

[RFC2544] methods have never been validated on a network path, even when that path is not part of a production network and carrying no other traffic. It is unknown whether the tests can be used to measure valid and reliable performance of a multi-device, multi-network path. It is possible that some of the tests may prove to be valid in some path scenarios, but that work has not been done or has not been shared with the IETF community. Thus, such testing is contra-indicated by the BMWG.

6. What to do without RFC 2544?

The IETF has addressed the problem of production network performance measurement by chartering a different working group: IP Performance Metrics (IPPM). This working group has developed a set of standard metrics to assess the quality, performance, and reliability of Internet packet transfer services. These metrics can be measured by network operators, end users, or independent testing groups. We note that some IPPM metrics differ from RFC 2544 metrics with similar names, and there is likely to be confusion if the details are ignored.

IPPM has not standardized methods for raw capacity measurement of Internet paths. Such testing needs to adequately consider the strong possibility for degradation to any other traffic that may be present due to congestion. There are no specific methods proposed for activation of a packet transfer service in IPPM.

Other standards may help to fill gaps in telecommunication service testing. For example, the IETF has many standards intended to assist with network operation, administration and maintenance (OAM), the ITU-T Study Group 12 has a recommendation on service activation test methodology.

The world will not spin off axis while waiting for appropriate and standardized methods to emerge from the consensus process.

7. Security Considerations

This Applicability Statement is also intended to help preserve the security of the Internet by clarifying that the scope of [RFC2544] and other BMWG memos are all limited to testing in laboratory ITE, thus avoiding accidental Denial of Service attacks or congestion due to high traffic volume test streams.

All Benchmarking activities are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the other constraints [RFC2544].

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the device under test/system under test (DUT/SUT).

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

8. IANA Considerations

This memo makes no requests of IANA, and hopes that IANA will leave it alone as well.

9. Acknowledgements

Thanks to Matt Zekauskas, Bill Cerveney, Barry Constantine, and Curtis Villamizar for reading and suggesting improvements for this memo.

10. References

10.1. Normative References

- [RFC1242] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [RFC2679] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Delay Metric for IPPM", RFC 2679, September 1999.

[RFC5180] Popoviciu, C., Hamza, A., Van de Velde, G., and D. Dugatkin, "IPv6 Benchmarking Methodology for Network Interconnect Devices", RFC 5180, May 2008.

10.2. Informative References

Authors' Addresses

Scott Bradner
Harvard University
29 Oxford St.
Cambridge, MA 02138
USA

Phone: +1 617 495 3864
Fax:
Email: sob@harvard.edu
URI: <http://www.sobco.com>

Kevin Dubray
Juniper Networks

Phone:
Fax:
Email: kdubray@juniper.net
URI:

Jim McQuaid
Turnip Video
6 Cobbleridge Court
Durham, North Carolina 27713
USA

Phone: +1 919-619-3220
Fax:
Email: jim@turnipvideo.com
URI: www.turnipvideo.com

Al Morton
AT&T Labs
200 Laurel Avenue South
Middletown,, NJ 07748
USA

Phone: +1 732 420 1571
Fax: +1 732 368 1192
Email: acmorton@att.com
URI: <http://home.comcast.net/~acmacm/>

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: March 17, 2012

M. Hamilton
BreakingPoint Systems
S. Banks
Cisco Systems
September 14, 2011

Benchmarking Methodology for Content-Aware Network Devices
draft-ietf-bmwg-ca-bench-meth-00

Abstract

This document defines a set of test scenarios and metrics that can be used to benchmark content-aware network devices. More specifically, these scenarios are designed to most accurately predict performance of these devices when subjected to relevant traffic patterns. This document will operate within the constraints of the Benchmarking Working Group charter, namely black box characterization in a laboratory environment.

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 <http://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 March 17, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	4
1.1. Requirements Language	5
2. Scope	5
3. Test Setup	5
3.1. Test Considerations	6
3.2. Clients and Servers	6
3.3. Traffic Generation Requirements	6
3.4. Discussion of Network Mathematics	6
3.5. Framework for Traffic Specification	8
3.6. Multiple Client/Server Testing	8
3.7. Device Configuration Considerations	8
3.7.1. Network Addressing	8
3.7.2. Network Address Translation	9
3.7.3. TCP Stack Considerations	9
3.7.4. Other Considerations	9
4. Benchmarking Tests	9
4.1. Maximum Application Flow Rate	9
4.1.1. Objective	10
4.1.2. Setup Parameters	10
4.1.2.1. Application-Layer Parameters	10
4.1.3. Procedure	10
4.1.4. Measurement	10
4.1.4.1. Maximum Application Flow Rate	10
4.1.4.2. Application Flow Duration	10
4.1.4.3. Packet Loss	11
4.1.4.4. Application Flow Latency	11
4.2. Application Throughput	11
4.2.1. Objective	11
4.2.2. Setup Parameters	11
4.2.2.1. Parameters	11
4.2.3. Procedure	11
4.2.4. Measurement	11
4.2.4.1. Maximum Throughput	11
4.2.4.2. Packet Loss	12
4.2.4.3. Maximum Application Flow Rate	12
4.2.4.4. Application Flow Duration	12
4.2.4.5. Packet Loss	12
4.2.4.6. Application Flow Latency	12
4.3. Malicious Traffic Handling	12
4.3.1. Objective	12
4.3.2. Setup Parameters	12

4.3.2.1. Parameters	12
4.3.3. Procedure	13
4.3.4. Measurement	13
4.4. Malformed Traffic Handling	13
4.4.1. Objective	13
4.4.2. Setup Parameters	13
4.4.3. Procedure	13
4.4.4. Measurement	14
5. Appendix A: Example Test Case	14
6. IANA Considerations	16
7. Security Considerations	16
8. References	16
8.1. Normative References	16
8.2. Informative References	17
Authors' Addresses	17

1. Introduction

Content-aware and deep packet inspection (DPI) device deployments have grown significantly in recent years. No longer are devices simply using Ethernet and IP headers to make forwarding decisions. This class of device now uses application-specific data to make these decisions. For example, a web-application firewall (WAF) may use search criteria upon the HTTP uniform resource indicator (URI)[1] to decide whether a HTTP GET method may traverse the network. In the case of lawful/legal intercept technology, a device could use the phone number within the Session Description Protocol[11] to determine whether a voice-over-IP phone may be allowed to connect. In addition to the development of entirely new classes of devices, devices that could historically be classified as 'stateless' or raw forwarding devices are now performing DPI functionality. Devices such as core and edge routers are now being developed with DPI functionality to make more intelligent routing and forwarding decisions.

The Benchmarking Working Group (BMWG) has historically produced Internet Drafts and Requests for Comment that are focused specifically on creating output metrics that are derived from a very specific and well-defined set of input parameters that are completely and unequivocally reproducible from test bed to test bed. The end goal of such methodologies is to, in the words of the RFC 2544 [2], reduce "specsmanship" in the industry. Existing BMWG work has certainly met this stated goal.

The BMWG has historically avoided the use of the term "realistic" throughout all of its drafts and RFCs. While this document will not explicitly use this term, the end goal of the terminology and methodology is to generate performance metrics that will be as close as possible to equivalent metrics in a production environment. It should be further noted that any metrics acquired from a production network SHOULD be captured according to the policies and procedures of the IPPM or PMOL working groups.

An explicit non-goal of this document is to replace existing methodology/terminology pairs such as RFC 2544 [2]/RFC 1242 [3] or RFC 3511 [4]/RFC 2647 [5]. The explicit goal of this document is to create a methodology and terminology pair that is more suited for modern devices while complementing the data acquired using existing BMWG methodologies. Existing BMWG work generally relies on completely repeatable input stimulus, expecting fully repeatable output. For unicast UDP streams, this makes complete sense. This document does not assume completely repeatable input stimulus. The nature of application-driven networks is such that a single dropped packet inherently changes the input stimulus from a network perspective. While application flows will be specified in great

detail, it simply is not practical to require totally repeatable input stimulus.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [6].

2. Scope

Content-aware devices take many forms, shapes and architectures. These devices are advanced network interconnect devices that inspect deep into the application payload of network data packets to do classification. They may be as simple as a firewall that uses application data inspection for rule set enforcement, or they may have advanced functionality such as performing protocol decoding and validation, anti-virus, anti-spam and even application exploit filtering.

This document is strictly focused on examining performance and robustness across a focused set of metrics: throughput(min/max/avg/sample std dev), transaction rates(successful/failed), application response times, concurrent flows, and unidirectional packet latency. None of the metrics captured through this methodology are specific to a device, nor do they characterize the functional behavior of those devices. The metrics are implementation independent. Functional testing of the DUT is outside the scope of this methodology.

Devices such as firewalls, intrusion detection and prevention devices, application delivery controllers, deep packet inspection devices, wide-area network(WAN) optimization devices, and unified threat management systems generally fall into the content-aware category. While this list may become obsolete, these are a subset of devices that fall under this scope of testing.

3. Test Setup

This document will be applicable to most test configurations and will not be confined to a discussion on specific test configurations. Since each DUT/SUT will have their own unique configuration, users SHOULD configure their device with the same parameters that would be used in the actual deployment of the device or a typical deployment, if the actual deployment is unknown. In order to improve repeatability, the DUT configuration SHOULD be published with the final benchmarking results. If available, command-line scripts used

to configured the DUT and any configuration information for the tester SHOULD be published with the final results

3.1. Test Considerations

3.2. Clients and Servers

Content-aware device testing SHOULD involve multiple clients and multiple servers. As with RFC 3511 [4], this methodology will use the terms virtual clients/servers because both the client and server will be represented by the tester and not actual clients/servers. Similarly defined in RFC 3511 [4], a data source may emulate multiple clients and/or servers within the context of the same test scenario. The test report SHOULD indicate the number of virtual clients/servers used during the test. IANA has reserved address ranges for laboratory characterization. These are defined for IPv4 and IPv6 by RFC 2544 Appendix C [2] and RFC 5180 Section 5.2 [7] respectively and SHOULD be consulted prior to testing.

3.3. Traffic Generation Requirements

The explicit purposes of content-aware devices vary widely, but these devices use information deeper inside the application flow to make decisions and classify traffic. This methodology will utilize traffic flows that resemble real application traffic without utilizing captures from live production networks. Application Flows, as defined in RFC 2722 [8] are able to be well-defined without simply referring to a network capture. An example traffic template is defined and listed in Section 5 of this document. A user of this methodology is free to utilize the example mix as provided in the appendix. If a user of this methodology understands the traffic patterns in their production network, that user SHOULD use the template provided in Section 5 to describe a traffic mix appropriate for their environment.

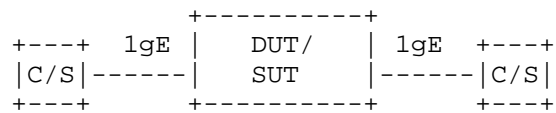
The test tool SHOULD be able to create application flows between every client and server, regardless of direction. The tester SHOULD be able to open TCP connections on multiple destination ports and SHOULD be able to direct UDP traffic to multiple destination ports.

3.4. Discussion of Network Mathematics

Prior to executing the methodology as outlined in the following sections, it is imperative to understand the implications of utilizing representative application flows for the traffic content of the benchmarking effort. One interesting aspect of utilizing application flows is that each flow is inherently different from every other application flow. The content of each flow will vary

from application to application, and in most cases, even varies within the same type of application flow. The following description of the methodology will individually benchmark every individual type and subset of application flow, prior to performing similar tests with a traffic mix as specified either by the example mix in Section 5, or as defined by the user of this methodology.

The purpose of this process is to ensure that any performance implications that are discovered during the mixed testing aren't due to the inherent physical network limitations. As an example of this phenomena, it is useful to examine a network device inserted into a single path, as illustrated in the following diagram.



Simple Inline DUT Configuration

Figure 1: Single Path Example

For the purpose of this discussion, let's take a theoretical application flow that utilizes UDP for the transport layer. Assume that the sample transaction we will be using to model this particular flow requires 10 UDP datagrams to complete the transaction. For simplicity, each datagram within the flow is exactly 64 bytes, including associated Ethernet, IP, and UDP overhead. With any network device, there are always three metrics which interact with each other: number of concurrent application flows, number of application flows per second, and layer-7 throughput.

Our example test bed is a single-path device connected with 1 gigabit Ethernet links. The purpose of this benchmark effort is to quantify the number of application flows per second that may be processed through our device under test. Let's assume that the result from our scenario is that the DUT is able to process 10,000 application flows per second. The question is whether that ceiling is the actual ceiling of the device, or if it is actually being limited by one of the other metrics. If we do the appropriate math, 10000 flows per second, with each flow at 640 total bytes means that we are achieving a throughput of roughly 49 Mbps. This is dramatically less than the 1 gigabit physical link we are using. We can conclude that 10,000 flows per second is in fact the performance limit of the device.

If we change the example slightly and increase the size of each datagram to 1312 bytes, then it becomes necessary to recompute the math. Assuming the same observed DUT limitation of 10,000 flows per

second, it must be ensured that this is an artifact of the DUT, and not of physical limitations. For each flow, we'll require 104,960 bits. 10,000 flows per second implies a throughput of roughly 1 Gbps. At this point, we cannot definitively answer whether the DUT is actually limited to 10,000 flows per second. If we are able to modify the scenario, and utilize 10 Gigabit interfaces, then perhaps the flow per second ceiling will be reached at a higher number than 10,000.

This example illustrates why a user of this methodology SHOULD benchmark each application variant individually to ensure that the cause of a measured limit is fully understood

3.5. Framework for Traffic Specification

The following table SHOULD be specified for each application flow variant.

- o Flow Size in Bits
- o Percentage of Aggregate Flows: 25%
- o Transport Protocol(s): TCP,UDP
- o Destination Port(s): 80

3.6. Multiple Client/Server Testing

In actual network deployments, connections are being established between multiple clients and multiple servers simultaneously. Device vendors have been known to optimize the operation of their devices for easily defined patterns. The connection sequence ordering scenarios a device will see on a network will likely be much less deterministic. In fact, many application flows have multiple layer 4 connections within a single flow, with client and server reversing roles. This methodology makes no assumptions about flow initiation sequence across multiple ports.

3.7. Device Configuration Considerations

The configuration of the DUT may have an effect on the observed results of the following methodology. A comprehensive, but certainly not exhaustive, list of potential considerations is listed below.

3.7.1. Network Addressing

The IANA has issued a range of IP addresses to the BMWG for purposes of benchmarking. Please refer to RFC 2544 [2] and RFC 5180 [7] for

more details.

3.7.2. Network Address Translation

Many content-aware devices are capable of performing Network Address Translation (NAT)[5]. If the final deployment of the DUT will have this functionality enabled, then the DUT SHOULD also have it enabled during the execution of this methodology. It MAY be beneficial to perform the test series in both modes in order to determine the performance differential when using NAT. The test report SHOULD indicate whether NAT was enabled during the testing process.

3.7.3. TCP Stack Considerations

The IETF has historically provided guidance and information on TCP stack considerations. This methodology is strictly focused on performance metrics at layers above 4, thus does not specifically define any TCP stack configuration parameters of either the tester or the DUTs. The TCP configuration of the tester SHOULD remain constant across all DUTs in order to ensure comparable results. While the following list of references is not exhaustive, each document contains a relevant discussion on TCP stack considerations.

Congestion control algorithms are discussed in Section 2 of RFC 3148 [9] with even more detailed references. TCP receive and congestion window sizes are discussed in detail in RFC 6349 [10].

3.7.4. Other Considerations

Various content-aware devices will have widely varying feature sets. In the interest of representative test results, the DUT features that will likely be enabled in the final deployment SHOULD be used. This methodology is not intended to advise on which features should be enabled, but to suggest using actual deployment configurations.

4. Benchmarking Tests

Each of the following benchmark scenarios SHOULD be run with each of the single application flow templates. Upon completion of all iterations, the mixed test SHOULD be completed, subject to the traffic mix as defined by the user.

4.1. Maximum Application Flow Rate

4.1.1. Objective

To determine the maximum rate through which a device is able to establish and complete application flows as defined by draft-ietf-bmwg-ca-bench-term-00.

4.1.2. Setup Parameters

The following parameters SHOULD be used and reported for all tests:

4.1.2.1. Application-Layer Parameters

For each application protocol in use during the test run, the table provided in Section 3.5 SHOULD be published.

4.1.3. Procedure

The test SHOULD generate application network traffic that meets the conditions of Section 3.3. The traffic pattern SHOULD begin with an application flow rate of 10% of expected maximum. The test SHOULD be configured to increase the attempt rate in units of 10% up through 110% of expected maximum. The duration of each loading phase SHOULD be at least 30 seconds. This test MAY be repeated, each subsequent iteration beginning at 5% of expected maximum and increasing session establishment rate to 10% more than the maximum observed from the previous test run.

This procedure MAY be repeated any number of times with the results being averaged together.

4.1.4. Measurement

The following metrics MAY be determined from this test, and SHOULD be observed for each application protocol within the traffic mix:

4.1.4.1. Maximum Application Flow Rate

The test tool SHOULD report the maximum rate at which application flows were completed, as defined by RFC 2647 [5], Section 3.7. This rate SHOULD be reported individually for each application protocol present within the traffic mix.

4.1.4.2. Application Flow Duration

The test tool SHOULD report the minimum, maximum and average application duration, as defined by RFC 2647 [5], Section 3.9. This duration SHOULD be reported individually for each application protocol present within the traffic mix.

4.1.4.3. Packet Loss

The test tool SHOULD report the number of flow packets lost or dropped from source to destination.

4.1.4.4. Application Flow Latency

The test tool SHOULD report the minimum, maximum and average amount of time an application flow member takes to traverse the DUT, as defined by RFC 1242 [3], Section 3.13. This rate SHOULD be reported individually for each application protocol present within the traffic mix.

4.2. Application Throughput

4.2.1. Objective

To determine the maximum rate through which a device is able to forward bits when using application flows as defined in the previous sections.

4.2.2. Setup Parameters

The following parameters SHOULD be used and reported for all tests:

4.2.2.1. Parameters

The same parameters as described in Section 4.1.2 SHOULD be used.

4.2.3. Procedure

This test will attempt to send application flows through the device at a flow rate of 30% of the maximum, as observed in Section 4.1. This procedure MAY be repeated with the results from each iteration averaged together.

4.2.4. Measurement

The following metrics MAY be determined from this test, and SHOULD be observed for each application protocol within the traffic mix:

4.2.4.1. Maximum Throughput

The test tool SHOULD report the minimum, maximum and average application throughput.

4.2.4.2. Packet Loss

The test tool SHOULD report the number of network packets lost or dropped from source to destination.

4.2.4.3. Maximum Application Flow Rate

The test tool SHOULD report the maximum rate at which application flows were completed, as defined by RFC 2647 [5], Section 3.7. This rate SHOULD be reported individually for each application protocol present within the traffic mix.

4.2.4.4. Application Flow Duration

The test tool SHOULD report the minimum, maximum and average application duration, as defined by RFC 2647 [5], Section 3.9. This duration SHOULD be reported individually for each application protocol present within the traffic mix.

4.2.4.5. Packet Loss

The test tool SHOULD report the number of flow packets lost or dropped from source to destination.

4.2.4.6. Application Flow Latency

The test tool SHOULD report the minimum, maximum and average amount of time an application flow member takes to traverse the DUT, as defined by RFC 1242 [3], Section 3.13. This rate SHOULD be reported individually for each application protocol present within the traffic mix.

4.3. Malicious Traffic Handling

4.3.1. Objective

To determine the effects on performance that malicious traffic may have on the DUT. While this test is not designed to characterize accuracy of detection or classification, it MAY be useful to record these measurements as specified below.

4.3.2. Setup Parameters

4.3.2.1. Parameters

The same parameters as described in Section 4.1.2 SHOULD be used.

Additionally, the following parameters SHOULD be used and reported

for all tests:

- o Attack List: A listing of the malicious traffic that was generated by the test.

4.3.3. Procedure

This test will utilize the procedures specified previously in Section 4.1.3 and Section 4.2.3. When performing the procedures listed previously, the tester should generate malicious traffic representative of the final network deployment. The mix of attacks MAY include software vulnerability exploits, network worms, back-door access attempts, network probes and other malicious traffic.

If a DUT can be run with and without the attack mitigation, both procedures SHOULD be run with and without the feature enabled on the DUT to determine the affects of the malicious traffic on the baseline metrics previously derived. If a DUT does not have active attack mitigation capabilities, this procedure SHOULD be run regardless. Certain malicious traffic could affect device performance even if the DUT does not actively inspect packet data for malicious traffic.

4.3.4. Measurement

The metrics specified by Section 4.1.4 and Section 4.2.4 SHOULD be determined from this test.

4.4. Malformed Traffic Handling

4.4.1. Objective

To determine the effects on performance and stability that malformed traffic may have on the DUT.

4.4.2. Setup Parameters

The same parameters SHOULD be used for Transport-Layer and Application Layer Parameters previously specified in Section 4.1.2 and Section 4.2.2.

4.4.3. Procedure

This test will utilize the procedures specified previously in Section 4.1.3 and Section 4.2.3. When performing the procedures listed previously, the tester should generate malformed traffic at all protocol layers. This is commonly known as fuzzed traffic. Fuzzing techniques generally modify portions of packets, including checksum errors, invalid protocol options, and improper protocol

conformance. This test SHOULD be run on a DUT regardless of whether it has built-in mitigation capabilities.

4.4.4. Measurement

For each protocol present in the traffic mix, the metrics specified by Section 4.1.4 and Section 4.2.4 MAY be determined. This data may be used to ascertain the effects of fuzzed traffic on the DUT.

5. Appendix A: Example Test Case

This appendix shows an example case of a protocol mix that may be used with this methodology.

Application Flow	Options	Value
Web 1kB	Flow Size (L7)	1kB
	Flow Percentage	15%
	Transport Protocol(s)	TCP
	Destination Port(s)	80
Web 10kB	Flow Size (L7)	10kB
	Flow Percentage	15%
	Transport Protocol(s)	TCP
	Destination Port(s)	80
Web 100kB	Flow Size (L7)	100kB
	Flow Percentage	15%
	Transport Protocol(s)	TCP
	Destination Port(s)	80
BitTorrent Movie Download	Flow Size (L7)	500 MB
	Flow Percentage	5%
	Transport Protocol(s)	TCP
	Destination Port(s)	6881-6889
SMTP Email	Flow Size (L7)	50 kB
	Flow Percentage	10%
	Transport Protocol(s)	TCP
	Destination Port(s)	25
IMAP Email	Flow Size (L7)	100 kB
	Flow Percentage	15%
	Transport Protocol(s)	TCP
	Destination Port(s)	143
DNS	Flow Size (L7)	2 kB
	Flow Percentage	10%
	Transport Protocol(s)	UDP
	Destination Port(s)	53
RTP	Flow Size (L7)	100 MB
	Flow Percentage	10%
	Transport Protocol(s)	UDP
	Destination Port(s)	20000-65535

Table 1: Sample Traffic Pattern

6. IANA Considerations

This memo includes no request to IANA.

All drafts are required to have an IANA considerations section (see the update of RFC 2434 [12] for a guide). If the draft does not require IANA to do anything, the section contains an explicit statement that this is the case (as above). If there are no requirements for IANA, the section will be removed during conversion into an RFC by the RFC Editor.

7. Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the other constraints RFC 2544 [2].

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network

8. References

8.1. Normative References

- [1] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.
- [2] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [3] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [4] Hickman, B., Newman, D., Tadjudin, S., and T. Martin, "Benchmarking Methodology for Firewall Performance", RFC 3511, April 2003.
- [5] Newman, D., "Benchmarking Terminology for Firewall Performance", RFC 2647, August 1999.
- [6] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

- [7] Popoviciu, C., Hamza, A., Van de Velde, G., and D. Dugatkin, "IPv6 Benchmarking Methodology for Network Interconnect Devices", RFC 5180, May 2008.
- [8] Brownlee, N., Mills, C., and G. Ruth, "Traffic Flow Measurement: Architecture", RFC 2722, October 1999.
- [9] Mathis, M. and M. Allman, "A Framework for Defining Empirical Bulk Transfer Capacity Metrics", RFC 3148, July 2001.
- [10] Constantine, B., Forget, G., Geib, R., and R. Schrage, "Framework for TCP Throughput Testing", RFC 6349, August 2011.

8.2. Informative References

- [11] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.
- [12] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

Authors' Addresses

Mike Hamilton
BreakingPoint Systems
Austin, TX 78717
US

Phone: +1 512 636 2303
Email: mhamilton@breakingpoint.com

Sarah Banks
Cisco Systems
San Jose, CA 95134
US

Email: sabanks@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 13, 2011

S. Poretsky
Allot Communications
B. Imhoff
Juniper Networks
K. Michielsen
Cisco Systems
February 16, 2011

Benchmarking Methodology for Link-State IGP Data Plane Route Convergence
draft-ietf-bmwg-igp-dataplane-conv-meth-23

Abstract

This document describes the methodology for benchmarking Link-State Interior Gateway Protocol (IGP) Route Convergence. The methodology is to be used for benchmarking IGP convergence time through externally observable (black box) data plane measurements. The methodology can be applied to any link-state IGP, such as IS-IS and OSPF.

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 <http://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 August 13, 2011.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1.	Introduction	5
1.1.	Motivation	5
1.2.	Factors for IGP Route Convergence Time	5
1.3.	Use of Data Plane for IGP Route Convergence Benchmarking	6
1.4.	Applicability and Scope	7
2.	Existing Definitions	7
3.	Test Topologies	8
3.1.	Test topology for local changes	8
3.2.	Test topology for remote changes	9
3.3.	Test topology for local ECMP changes	11
3.4.	Test topology for remote ECMP changes	11
3.5.	Test topology for Parallel Link changes	12
4.	Convergence Time and Loss of Connectivity Period	13
4.1.	Convergence Events without instant traffic loss	14
4.2.	Loss of Connectivity (LoC)	16
5.	Test Considerations	17
5.1.	IGP Selection	17
5.2.	Routing Protocol Configuration	17
5.3.	IGP Topology	17
5.4.	Timers	18
5.5.	Interface Types	18
5.6.	Offered Load	19
5.7.	Measurement Accuracy	20
5.8.	Measurement Statistics	20
5.9.	Tester Capabilities	20
6.	Selection of Convergence Time Benchmark Metrics and Methods	21
6.1.	Loss-Derived Method	21
6.1.1.	Tester capabilities	21
6.1.2.	Benchmark Metrics	21
6.1.3.	Measurement Accuracy	21
6.2.	Rate-Derived Method	22
6.2.1.	Tester Capabilities	22
6.2.2.	Benchmark Metrics	23
6.2.3.	Measurement Accuracy	23
6.3.	Route-Specific Loss-Derived Method	24
6.3.1.	Tester Capabilities	24
6.3.2.	Benchmark Metrics	24
6.3.3.	Measurement Accuracy	24
7.	Reporting Format	24
8.	Test Cases	26
8.1.	Interface Failure and Recovery	27
8.1.1.	Convergence Due to Local Interface Failure and Recovery	27
8.1.2.	Convergence Due to Remote Interface Failure and Recovery	28

8.1.3.	Convergence Due to ECMP Member Local Interface Failure and Recovery	30
8.1.4.	Convergence Due to ECMP Member Remote Interface Failure and Recovery	31
8.1.5.	Convergence Due to Parallel Link Interface Failure and Recovery	32
8.2.	Other Failures and Recoveries	33
8.2.1.	Convergence Due to Layer 2 Session Loss and Recovery	33
8.2.2.	Convergence Due to Loss and Recovery of IGP Adjacency	34
8.2.3.	Convergence Due to Route Withdrawal and Re-advertisement	35
8.3.	Administrative changes	37
8.3.1.	Convergence Due to Local Interface Administrative Changes	37
8.3.2.	Convergence Due to Cost Change	38
9.	Security Considerations	39
10.	IANA Considerations	40
11.	Acknowledgements	40
12.	References	40
12.1.	Normative References	40
12.2.	Informative References	41
	Authors' Addresses	42

1. Introduction

1.1. Motivation

Convergence time is a critical performance parameter. Service Providers use IGP convergence time as a key metric of router design and architecture. Fast network convergence can be optimally achieved through deployment of fast converging routers. Customers of Service Providers use packet loss due to Interior Gateway Protocol (IGP) convergence as a key metric of their network service quality. IGP route convergence is a Direct Measure of Quality (DMOQ) when benchmarking the data plane. The fundamental basis by which network users and operators benchmark convergence is packet loss and other packet impairments, which are externally observable events having direct impact on their application performance. For this reason it is important to develop a standard methodology for benchmarking link-state IGP convergence time through externally observable (black-box) data plane measurements. All factors contributing to convergence time are accounted for by measuring on the data plane.

1.2. Factors for IGP Route Convergence Time

There are four major categories of factors contributing to the measured IGP convergence time. As discussed in [Vi02], [Ka02], [Fi02], [Al00], [Al02], and [Fr05], these categories are Event Detection, Shortest Path First (SPF) Processing, Link State Advertisement (LSA) / Link State Packet (LSP) Advertisement, and Forwarding Information Base (FIB) Update. These have numerous components that influence the convergence time, including but not limited to the list below:

- o Event Detection
 - * Physical Layer failure/recovery indication time
 - * Layer 2 failure/recovery indication time
 - * IGP Hello Dead Interval
- o SPF Processing
 - * SPF Delay Time
 - * SPF Hold time
 - * SPF Execution time

- o LSA/LSP Advertisement
 - * LSA/LSP Generation time
 - * LSA/LSP Flood Packet Pacing
 - * LSA/LSP Retransmission Packet Pacing
- o FIB Update
 - * Tree Build time
 - * Hardware Update time
- o Increased Forwarding Delay due to Queueing

The contribution of each of these factors listed above will vary with each router vendors' architecture and IGP implementation. Routers may have a centralized forwarding architecture, in which one forwarding table is calculated and referenced for all arriving packets, or a distributed forwarding architecture, in which the central forwarding table is calculated and distributed to the interfaces for local look-up as packets arrive. The distributed forwarding tables are typically maintained in hardware.

The variation in router architecture and implementation necessitates the design of a convergence test that considers all of these components contributing to convergence time and is independent of the Device Under Test (DUT) architecture and implementation. The benefit of designing a test for these considerations is that it enables black-box testing in which knowledge of the routers' internal implementation is not required. It is then possible to make valid use of the convergence benchmarking metrics when comparing routers from different vendors.

Convergence performance is tightly linked to the number of tasks a router has to deal with. As the most impacting tasks are mainly related to the control plane and the data plane, the more the DUT is stressed as in a live production environment, the closer performance measurement results match the ones that would be observed in a live production environment.

1.3. Use of Data Plane for IGP Route Convergence Benchmarking

Customers of Service Providers use packet loss and other packet impairments as metrics to calculate convergence time. Packet loss and other packet impairments are externally observable events having direct impact on customers' application performance. For this reason

it is important to develop a standard router benchmarking methodology that is a Direct Measure of Quality (DMOQ) for measuring IGP convergence. An additional benefit of using packet loss for calculation of IGP Route Convergence time is that it enables black-box tests to be designed. Data traffic can be offered to the Device Under Test (DUT), an emulated network event can be forced to occur, and packet loss and other impaired packets can be externally measured to calculate the convergence time. Knowledge of the DUT architecture and IGP implementation is not required. There is no need to rely on the DUT to produce the test results. There is no need to build intrusive test harnesses for the DUT. All factors contributing to convergence time are accounted for by measuring on the dataplane.

Other work of the Benchmarking Methodology Working Group (BMWG) focuses on characterizing single router control plane convergence. See [Ma05], [Ma05t], and [Ma05c].

1.4. Applicability and Scope

The methodology described in this document can be applied to IPv4 and IPv6 traffic and link-state IGPs such as IS-IS [Ca90][Ho08], OSPF [Mo98][Co08], and others. IGP adjacencies established over any kind of tunnel (such as Traffic Engineering tunnels) are outside the scope of this document. Convergence time benchmarking in topologies with non point-to-point IGP adjacencies will be covered in a later document. Convergence from Bidirectional Forwarding Detection (BFD) is outside the scope of this document. Non-Stop Forwarding (NSF), Non-Stop Routing (NSR), Graceful Restart (GR), or any other High Availability mechanism are outside the scope of this document. Fast reroute mechanisms such as IP Fast-Reroute [Sh10i] or MPLS Fast-Reroute [Pa05] are outside the scope of this document.

2. Existing Definitions

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 BCP 14, RFC 2119 [Br97]. RFC 2119 defines the use of these key words to help make the intent of standards track documents as clear as possible. While this document uses these keywords, this document is not a standards track document.

This document uses much of the terminology defined in [Pol1t]. For any conflicting content, this document supersedes [Pol1t]. This document uses existing terminology defined in other documents issued by the Benchmarking Methodology Working Group (BMWG). Examples include, but are not limited to:

Throughput	[Ref.[Br91], section 3.17]
Device Under Test (DUT)	[Ref.[Ma98], section 3.1.1]
System Under Test (SUT)	[Ref.[Ma98], section 3.1.2]
Out-of-Order Packet	[Ref.[Po06], section 3.3.4]
Duplicate Packet	[Ref.[Po06], section 3.3.5]
Stream	[Ref.[Po06], section 3.3.2]
Loss Period	[Ref.[Ko02], section 4]
Forwarding Delay	[Ref.[Po06], section 3.2.4]
IP Packet Delay Variation (IPDV)	[Ref.[De02], section 1.2]

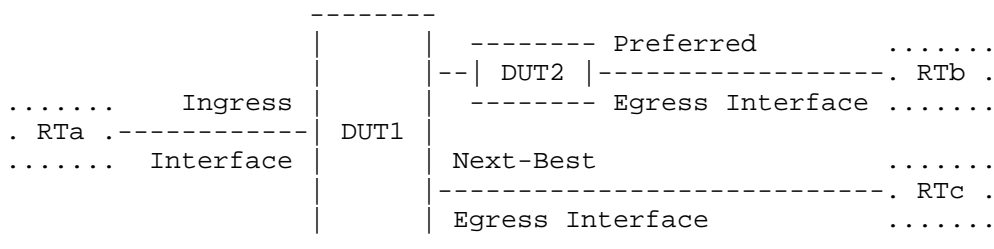


Figure 3: IGP convergence test topology for remote changes

Figure 4 shows the test topology to measure IGP convergence time due to remote Convergence Events with a non-ECMP Preferred Egress Interface and ECMP Next-Best Egress Interfaces (Section 8.1.2). In this topology the two routers DUT1 and DUT2 are considered System Under Test (SUT) and MUST be identically configured devices of the same model. Router DUT1 is configured with the Next-Best Egress Interface an ECMP set of interfaces. The Preferred Egress Interface of DUT1 is not a member of an ECMP set. The Tester emulates $N+2$ neighbor routers ($N>0$), one for the Ingress Interface (RTa), one for DUT2 (RTb) and one for each member of the ECMP set (RTc1...RTcN). IGP adjacencies MUST be established between Tester and SUT, one on each interface of SUT. For this purpose each of the $N+2$ routers emulated by the Tester establishes one adjacency with the SUT. In this topology there is a possibility of a packet forwarding loop that may occur transiently between DUT1 and DUT2 during convergence (micro-loop, see [Sh10]). When the test specifies to observe the Next-Best Egress Interface statistics, the combined statistics for all members of the ECMP set should be observed.

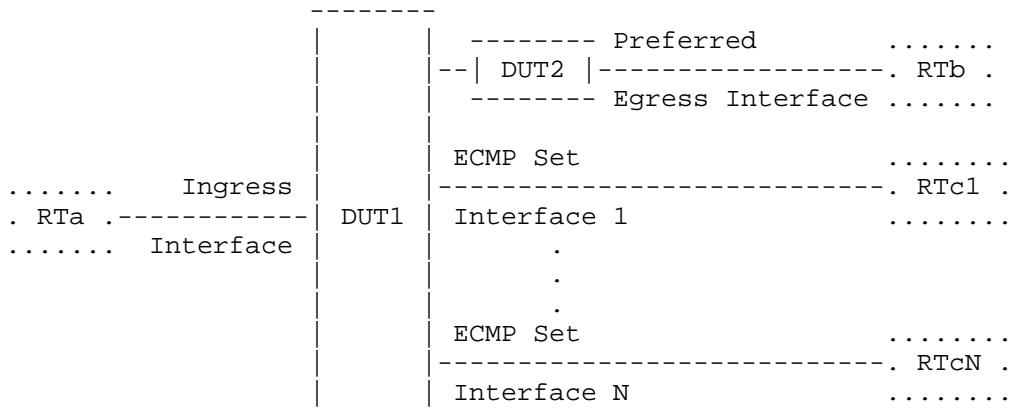


Figure 4: IGP convergence test topology for remote changes with non-

ECMP to ECMP convergence

3.3. Test topology for local ECMP changes

Figure 5 shows the test topology to measure IGP convergence time due to local Convergence Events of a member of an Equal Cost Multipath (ECMP) set (Section 8.1.3). In this topology, the DUT is configured with each egress interface as a member of a single ECMP set and the Tester emulates N+1 next-hop routers, one for the Ingress Interface (RTa) and one for each member of the ECMP set (RTb1...RTbN). IGP adjacencies MUST be established between Tester and DUT, one on the Ingress Interface and one on each member of the ECMP set. For this purpose each of the N+1 routers emulated by the Tester establishes one adjacency with the DUT. When the test specifies to observe the Next-Best Egress Interface statistics, the combined statistics for all ECMP members except the one affected by the Convergence Event, should be observed.

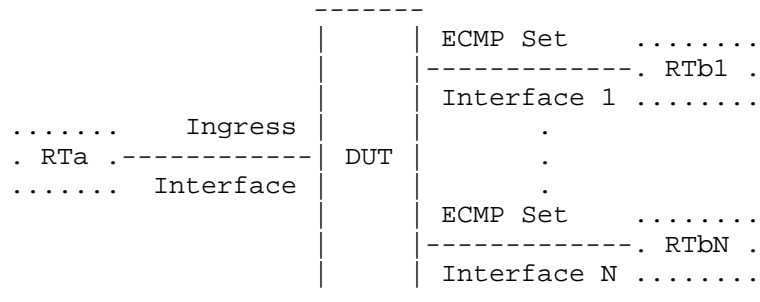


Figure 5: IGP convergence test topology for local ECMP changes

3.4. Test topology for remote ECMP changes

Figure 6 shows the test topology to measure IGP convergence time due to remote Convergence Events of a member of an Equal Cost Multipath (ECMP) set (Section 8.1.4). In this topology the two routers DUT1 and DUT2 are considered System Under Test (SUT) and MUST be identically configured devices of the same model. Router DUT1 is configured with each egress interface as a member of a single ECMP set and the Tester emulates N+1 neighbor routers (N>0), one for the Ingress Interface (RTa) and one for each member of the ECMP set (RTb1...RTbN). IGP adjacencies MUST be established between Tester and SUT, one on each interface of SUT. For this purpose each of the N+1 routers emulated by the Tester establishes one adjacency with the SUT (N-1 emulated routers are adjacent to DUT1 egress interfaces, one emulated router is adjacent to DUT1 Ingress Interface, and one emulated router is adjacent to DUT2). In this topology there is a

Figure 7: IGP convergence test topology for Parallel Link changes

At least one condition needs to be fulfilled for Route Convergence time to be equal to Route Loss of Connectivity Period. The condition is that the Convergence Event causes an instantaneous traffic loss for the measured route. A fiber cut on the Preferred Egress Interface is an example of such a Convergence Event.

A second condition applies to Route Convergence time measurements based on Connectivity Packet Loss [Pollt]. This second condition is that there is only a single Loss Period during Route Convergence. For the testcases described in this document this is expected to be the case.

4.1. Convergence Events without instant traffic loss

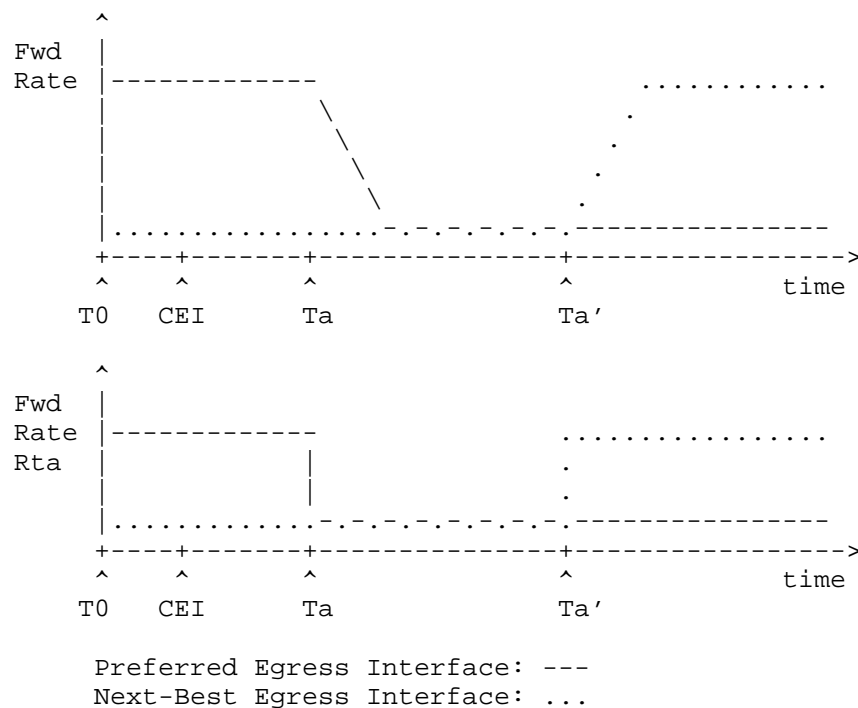
To measure convergence time benchmarks for Convergence Events caused by a Tester, such as an IGP cost change, the Tester MAY start to discard all traffic received from the Preferred Egress Interface at the Convergence Event Instant, or MAY separately observe packets received from the Preferred Egress Interface prior to the Convergence Event Instant. This way these Convergence Events can be treated the same as Convergence Events that cause instantaneous traffic loss.

To measure convergence time benchmarks without instantaneous traffic loss (either real or induced by the Tester) at the Convergence Event Instant, such as a reversion of a link failure Convergence Event, the Tester SHALL only observe packet statistics on the Next-Best Egress Interface. If using the Rate-Derived method to benchmark convergence times for such Convergence Events, the Tester MUST collect a timestamp at the Convergence Event Instant. If using a loss-derived method to benchmark convergence times for such Convergence Events, the Tester MUST measure the period in time between the Start Traffic Instant and the Convergence Event Instant. To measure this period in time the Tester can collect timestamps at the Start Traffic Instant and the Convergence Event Instant.

The Convergence Event Instant together with the receive rate observations on the Next-Best Egress Interface allow to derive the convergence time benchmarks using the Rate-Derived Method [Pollt].

By observing packets on the Next-Best Egress Interface only, the observed Impaired Packet count is the number of Impaired Packets between Traffic Start Instant and Convergence Recovery Instant. To measure convergence times using a loss-derived method, the Impaired Packet count between the Convergence Event Instant and the Convergence Recovery Instant is needed. The time between Traffic Start Instant and Convergence Event Instant must be accounted for. An example may clarify this.

Figure 8 illustrates a Convergence Event without instantaneous traffic loss for all routes. The top graph shows the Forwarding Rate over all routes, the bottom graph shows the Forwarding Rate for a single route Rta. Some time after the Convergence Event Instant, Forwarding Rate observed on the Preferred Egress Interface starts to decrease. In the example, route Rta is the first route to experience packet loss at time Ta. Some time later, the Forwarding Rate observed on the Next-Best Egress Interface starts to increase. In the example, route Rta is the first route to complete convergence at time Ta'.



With T0 the Start Traffic Instant; CEI the Convergence Event Instant; Ta the time instant packet loss for route Rta starts; Ta' the time instant packet impairment for route Rta ends.

Figure 8

If only packets received on the Next-Best Egress Interface are observed, the duration of the loss period for route Rta can be calculated from the received packets as in Equation 1. Since the Convergence Event Instant is the start time for convergence time measurement, the period in time between T0 and CEI needs to be subtracted from the calculated result to become the convergence time,

as in Equation 2.

```

Next-Best Egress Interface loss period
= (packets transmitted
  - packets received from Next-Best Egress Interface) / tx rate
= Ta' - T0

```

Equation 1

```

convergence time
= Next-Best Egress Interface loss period - (CEI - T0)
= Ta' - CEI

```

Equation 2

4.2. Loss of Connectivity (LoC)

Route Loss of Connectivity Period SHOULD be measured using the Route-Specific Loss-Derived Method. Since the start instant and end instant of the Route Loss of Connectivity Period can be different for each route, these can not be accurately derived by only observing global statistics over all routes. An example may clarify this.

Following a Convergence Event, route Rta is the first route for which packet impairment starts, the Route Loss of Connectivity Period for route Rta starts at time Ta. Route Rtb is the last route for which packet impairment starts, the Route Loss of Connectivity Period for route Rtb starts at time Tb with Tb>Ta.

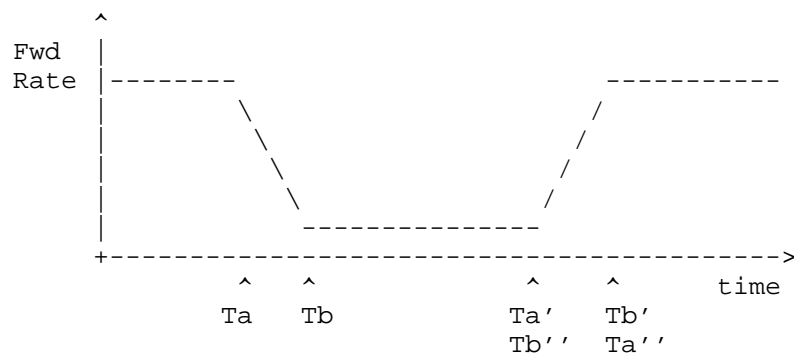


Figure 9: Example Route Loss Of Connectivity Period

If the DUT implementation were such that route Rta would be the first route for which traffic loss ends at time Ta' (with Ta'>Tb) and route Rtb would be the last route for which traffic loss ends at time Tb' (with Tb'>Ta'). By only observing global traffic statistics over all

routes, the minimum Route Loss of Connectivity Period would be measured as $Ta'-Ta$. The maximum calculated Route Loss of Connectivity Period would be $Tb'-Ta$. The real minimum and maximum Route Loss of Connectivity Periods are $Ta'-Ta$ and $Tb'-Tb$. Illustrating this with the numbers $Ta=0$, $Tb=1$, $Ta'=3$, and $Tb'=5$, would give a Loss of Connectivity Period between 3 and 5 derived from the global traffic statistics, versus the real Loss of Connectivity Period between 3 and 4.

If the DUT implementation were such that route R_{tb} would be the first for which packet loss ends at time Tb'' and route R_{ta} would be the last for which packet impairment ends at time Ta'' , then the minimum and maximum Route Loss of Connectivity Periods derived by observing only global traffic statistics would be $Tb''-Ta$, and $Ta''-Ta$. The real minimum and maximum Route Loss of Connectivity Periods are $Tb''-Tb$ and $Ta''-Ta$. Illustrating this with the numbers $Ta=0$, $Tb=1$, $Ta''=5$, $Tb''=3$, would give a Loss of Connectivity Period between 3 and 5 derived from the global traffic statistics, versus the real Loss of Connectivity Period between 2 and 5.

The two implementation variations in the above example would result in the same derived minimum and maximum Route Loss of Connectivity Periods when only observing the global packet statistics, while the real Route Loss of Connectivity Periods are different.

5. Test Considerations

5.1. IGP Selection

The test cases described in Section 8 can be used for link-state IGPs, such as IS-IS or OSPF. The IGP convergence time test methodology is identical.

5.2. Routing Protocol Configuration

The obtained results for IGP convergence time may vary if other routing protocols are enabled and routes learned via those protocols are installed. IGP convergence times SHOULD be benchmarked without routes installed from other protocols. Any enabled IGP routing protocol extension (such as extensions for Traffic Engineering) and any enabled IGP routing protocol security mechanism must be reported with the results.

5.3. IGP Topology

The Tester emulates a single IGP topology. The DUT establishes IGP adjacencies with one or more of the emulated routers in this single

IGP topology emulated by the Tester. See test topology details in Section 3. The emulated topology SHOULD only be advertised on the DUT egress interfaces.

The number of IGP routes and number of nodes in the topology, and the type of topology will impact the measured IGP convergence time. To obtain results similar to those that would be observed in an operational network, it is RECOMMENDED that the number of installed routes and nodes closely approximate that of the network (e.g. thousands of routes with tens or hundreds of nodes).

The number of areas (for OSPF) and levels (for IS-IS) can impact the benchmark results.

5.4. Timers

There are timers that may impact the measured IGP convergence times. The benchmark metrics MAY be measured at any fixed values for these timers. To obtain results similar to those that would be observed in an operational network, it is RECOMMENDED to configure the timers with the values as configured in the operational network.

Examples of timers that may impact measured IGP convergence time include, but are not limited to:

- Interface failure indication

- IGP hello timer

- IGP dead-interval or hold-timer

- Link State Advertisement (LSA) or Link State Packet (LSP) generation delay

- LSA or LSP flood packet pacing

- route calculation delay

5.5. Interface Types

All test cases in this methodology document can be executed with any interface type. The type of media may dictate which test cases may be executed. Each interface type has a unique mechanism for detecting link failures and the speed at which that mechanism operates will influence the measurement results. All interfaces MUST be the same media and Throughput [Br91][Br99] for each test case. All interfaces SHOULD be configured as point-to-point.

5.6. Offered Load

The Throughput of the device, as defined in [Br91] and benchmarked in [Br99] at a fixed packet size, needs to be determined over the preferred path and over the next-best path. The Offered Load SHOULD be the minimum of the measured Throughput of the device over the primary path and over the backup path. The packet size is selectable and MUST be recorded. Packet size is measured in bytes and includes the IP header and payload.

The destination addresses for the Offered Load MUST be distributed such that all routes or a statistically representative subset of all routes are matched and each of these routes is offered an equal share of the Offered Load. It is RECOMMENDED to send traffic matching all routes, but a statistically representative subset of all routes can be used if required.

Splitting traffic flows across multiple paths (as with ECMP or Parallel Link sets) is in general done by hashing on various fields on the IP or contained headers. The hashing is typically based on the IP source and destination addresses, the protocol ID, and higher-layer flow-dependent fields such as TCP/UDP ports. In practice, within a network core, the hashing is based mainly or exclusively on the IP source and destination addresses. Knowledge of the hashing algorithm used by the DUT is not always possible beforehand, and would violate the black-box spirit of this document. Therefore it is RECOMMENDED to use a randomly distributed range of source and destination IP addresses, protocol IDs, and higher-layer flow-dependent fields for the packets of the Offered Load (see also [Ne07]). The content of the Offered Load MUST remain the same during the test. It is RECOMMENDED to repeat a test multiple times with different random ranges of the header fields such that convergence time benchmarks are measured for different distributions of traffic over the available paths.

In the Remote Interface failure testcases using topologies 3, 4, and 6 there is a possibility of a packet forwarding loop that may occur transiently between DUT1 and DUT2 during convergence (micro-loop, see [Sh10]). The Time To Live (TTL) or Hop Limit value of the packets sent by the Tester may influence the benchmark measurements since it determines which device in the topology may send an ICMP Time Exceeded Message for looped packets.

The duration of the Offered Load MUST be greater than the convergence time plus the Sustained Convergence Validation Time.

Offered load should send a packet to each destination before sending another packet to the same destination. It is RECOMMENDED that the

packets be transmitted in a round-robin fashion with a uniform interpacket delay.

5.7. Measurement Accuracy

Since Impaired Packet count is observed to measure the Route Convergence Time, the time between two successive packets offered to each individual route is the highest possible accuracy of any Impaired Packet based measurement. The higher the traffic rate offered to each route the higher the possible measurement accuracy.

Also see Section 6 for method-specific measurement accuracy.

5.8. Measurement Statistics

The benchmark measurements may vary for each trial, due to the statistical nature of timer expirations, cpu scheduling, etc. Evaluation of the test data must be done with an understanding of generally accepted testing practices regarding repeatability, variance and statistical significance of a small number of trials.

5.9. Tester Capabilities

It is RECOMMENDED that the Tester used to execute each test case have the following capabilities:

1. Ability to establish IGP adjacencies and advertise a single IGP topology to one or more peers.
2. Ability to measure Forwarding Delay, Duplicate Packets and Out-of-Order Packets.
3. An internal time clock to control timestamping, time measurements, and time calculations.
4. Ability to distinguish traffic load received on the Preferred and Next-Best Interfaces [Pollt].
5. Ability to disable or tune specific Layer-2 and Layer-3 protocol functions on any interface(s).

The Tester MAY be capable to make non-data plane convergence observations and use those observations for measurements. The Tester MAY be capable to send and receive multiple traffic Streams [Po06].

Also see Section 6 for method-specific capabilities.

6. Selection of Convergence Time Benchmark Metrics and Methods

Different convergence time benchmark methods MAY be used to measure convergence time benchmark metrics. The Tester capabilities are important criteria to select a specific convergence time benchmark method. The criteria to select a specific benchmark method include, but are not limited to:

Tester capabilities:	Sampling Interval, number of Stream statistics to collect
Measurement accuracy:	Sampling Interval, Offered Load, number of routes
Test specification:	number of routes
DUT capabilities:	Throughput, IP Packet Delay Variation

6.1. Loss-Derived Method

6.1.1. Tester capabilities

To enable collecting statistics of Out-of-Order Packets per flow (See [Th00], Section 3) the Offered Load SHOULD consist of multiple Streams [Po06] and each Stream SHOULD consist of a single flow . If sending multiple Streams, the measured traffic statistics for all Streams MUST be added together.

In order to verify Full Convergence completion and the Sustained Convergence Validation Time, the Tester MUST measure Forwarding Rate each Packet Sampling Interval.

The total number of Impaired Packets between the start of the traffic and the end of the Sustained Convergence Validation Time is used to calculate the Loss-Derived Convergence Time.

6.1.2. Benchmark Metrics

The Loss-Derived Method can be used to measure the Loss-Derived Convergence Time, which is the average convergence time over all routes, and to measure the Loss-Derived Loss of Connectivity Period, which is the average Route Loss of Connectivity Period over all routes.

6.1.3. Measurement Accuracy

The actual value falls within the accuracy interval $[-(\text{number of destinations/Offered Load}), +(\text{number of destinations/Offered Load})]$ around the value as measured using the Loss-Derived Method.

6.2. Rate-Derived Method

6.2.1. Tester Capabilities

To enable collecting statistics of Out-of-Order Packets per flow (See [Th00], Section 3) the Offered Load SHOULD consist of multiple Streams [Po06] and each Stream SHOULD consist of a single flow . If sending multiple Streams, the measured traffic statistics for all Streams MUST be added together.

The Tester measures Forwarding Rate each Sampling Interval. The Packet Sampling Interval influences the observation of the different convergence time instants. If the Packet Sampling Interval is large compared to the time between the convergence time instants, then the different time instants may not be easily identifiable from the Forwarding Rate observation. The presence of IP Packet Delay Variation (IPDV) [De02] may cause fluctuations of the Forwarding Rate observation and can prevent correct observation of the different convergence time instants.

The Packet Sampling Interval MUST be larger than or equal to the time between two consecutive packets to the same destination. For maximum accuracy the value for the Packet Sampling Interval SHOULD be as small as possible, but the presence of IPDV may enforce using a larger Packet Sampling Interval. The Packet Sampling Interval MUST be reported.

IPDV causes fluctuations in the number of received packets during each Packet Sampling Interval. To account for the presence of IPDV in determining if a convergence instant has been reached, Forwarding Delay SHOULD be observed during each Packet Sampling Interval. The minimum and maximum number of packets expected in a Packet Sampling Interval in presence of IPDV can be calculated with Equation 3.

number of packets expected in a Packet Sampling Interval
in presence of IP Packet Delay Variation
= expected number of packets without IP Packet Delay Variation
+/- ((maxDelay - minDelay) * Offered Load)
with minDelay and maxDelay the minimum resp. maximum Forwarding Delay
of packets received during the Packet Sampling Interval

Equation 3

To determine if a convergence instant has been reached the number of packets received in a Packet Sampling Interval is compared with the range of expected number of packets calculated in Equation 3.

6.2.2. Benchmark Metrics

The Rate-Derived Method SHOULD be used to measure First Route Convergence Time and Full Convergence Time. It SHOULD NOT be used to measure Loss of Connectivity Period (see Section 4).

6.2.3. Measurement Accuracy

The measurement accuracy interval of the Rate-Derived Method depends on the metric being measured or calculated and the characteristics of the related transition. IP Packet Delay Variation (IPDV) [De02] adds uncertainty to the amount of packets received in a Packet Sampling Interval and this uncertainty adds to the measurement error. The effect of IPDV is not accounted for in the calculation of the accuracy intervals below. IPDV is of importance for the convergence instants where a variation in Forwarding Rate needs to be observed (Convergence Recovery Instant and for topologies with ECMP also Convergence Event Instant and First Route Convergence Instant).

If the Convergence Event Instant is observed on the dataplane using the Rate Derived Method, it needs to be instantaneous for all routes (see Section 4.1). The actual value of the Convergence Event Instant falls within the accuracy interval $[-(\text{Packet Sampling Interval} + 1/\text{Offered Load}), +0]$ around the value as measured using the Rate-Derived Method.

If the Convergence Recovery Transition is non-instantaneous for all routes then the actual value of the First Route Convergence Instant falls within the accuracy interval $[-(\text{Packet Sampling Interval} + \text{time between two consecutive packets to the same destination}), +0]$ around the value as measured using the Rate-Derived Method, and the actual value of the Convergence Recovery Instant falls within the accuracy interval $[-(2 * \text{Packet Sampling Interval}), -(\text{Packet Sampling Interval} - \text{time between two consecutive packets to the same destination})]$ around the value as measured using the Rate-Derived Method.

The term "time between two consecutive packets to the same destination" is added in the above accuracy intervals since packets are sent in a particular order to all destinations in a stream and when part of the routes experience packet loss, it is unknown where in the transmit cycle packets to these routes are sent. This uncertainty adds to the error.

The accuracy intervals of the derived metrics First Route Convergence Time and Rate-Derived Convergence Time are calculated from the above convergence instants accuracy intervals. The actual value of First Route Convergence Time falls within the accuracy interval $[-(\text{Packet Sampling Interval} + \text{time between two consecutive packets to the same destination}), +0]$ around the value as measured using the Rate-Derived Method.

destination), $+(Packet\ Sampling\ Interval + 1/Offered\ Load)$] around the calculated value. The actual value of Rate-Derived Convergence Time falls within the accuracy interval $[-(2 * Packet\ Sampling\ Interval), +(time\ between\ two\ consecutive\ packets\ to\ the\ same\ destination + 1/Offered\ Load)]$ around the calculated value.

6.3. Route-Specific Loss-Derived Method

6.3.1. Tester Capabilities

The Offered Load consists of multiple Streams. The Tester MUST measure Impaired Packet count for each Stream separately.

In order to verify Full Convergence completion and the Sustained Convergence Validation Time, the Tester MUST measure Forwarding Rate each Packet Sampling Interval. This measurement at each Packet Sampling Interval MAY be per Stream.

Only the total number of Impaired Packets measured per Stream at the end of the Sustained Convergence Validation Time is used to calculate the benchmark metrics with this method.

6.3.2. Benchmark Metrics

The Route-Specific Loss-Derived Method SHOULD be used to measure Route-Specific Convergence Times. It is the RECOMMENDED method to measure Route Loss of Connectivity Period.

Under the conditions explained in Section 4, First Route Convergence Time and Full Convergence Time as benchmarked using Rate-Derived Method, may be equal to the minimum resp. maximum of the Route-Specific Convergence Times.

6.3.3. Measurement Accuracy

The actual value falls within the accuracy interval $[-(number\ of\ destinations/Offered\ Load), +(number\ of\ destinations/Offered\ Load)]$ around the value as measured using the Route-Specific Loss-Derived Method.

7. Reporting Format

For each test case, it is RECOMMENDED that the reporting tables below be completed and all time values SHOULD be reported with a sufficiently high resolution.

Parameter	Units
-----	-----
Test Case	test case number
Test Topology	Test Topology Figure number
IGP	(IS-IS, OSPF, other)
Interface Type	(GigE, POS, ATM, other)
Packet Size offered to DUT	bytes
Offered Load	packets per second
IGP Routes advertised to DUT	number of IGP routes
Nodes in emulated network	number of nodes
Number of Parallel or ECMP links	number of links
Number of Routes measured	number of routes
Packet Sampling Interval on Tester	seconds
Forwarding Delay Threshold	seconds
Timer Values configured on DUT:	
Interface failure indication delay	seconds
IGP Hello Timer	seconds
IGP Dead-Interval or hold-time	seconds
LSA/LSP Generation Delay	seconds
LSA/LSP Flood Packet Pacing	seconds
LSA/LSP Retransmission Packet Pacing	seconds
route calculation Delay	seconds

Test Details:

Describe the IGP extensions and IGP security mechanisms that are configured on the DUT.

Describe how the various fields on the IP and contained headers for the packets for the Offered Load are generated (Section 5.6).

If the Offered Load matches a subset of routes, describe how this subset is selected.

Describe how the Convergence Event is applied; does it cause instantaneous traffic loss or not?

The table below should be completed for the initial Convergence Event and the reversion Convergence Event.

Parameter	Units
-----	-----
Convergence Event	(initial or reversion)
Traffic Forwarding Metrics:	
Total number of packets offered to DUT	number of Packets
Total number of packets forwarded by DUT	number of Packets
Connectivity Packet Loss	number of Packets
Convergence Packet Loss	number of Packets
Out-of-Order Packets	number of Packets
Duplicate Packets	number of Packets
excessive Forwarding Delay Packets	number of Packets
Convergence Benchmarks:	
Rate-Derived Method:	
First Route Convergence Time	seconds
Full Convergence Time	seconds
Loss-Derived Method:	
Loss-Derived Convergence Time	seconds
Route-Specific Loss-Derived Method:	
Route-Specific Convergence Time[n]	array of seconds
Minimum Route-Specific Convergence Time	seconds
Maximum Route-Specific Convergence Time	seconds
Median Route-Specific Convergence Time	seconds
Average Route-Specific Convergence Time	seconds
Loss of Connectivity Benchmarks:	
Loss-Derived Method:	
Loss-Derived Loss of Connectivity Period	seconds
Route-Specific Loss-Derived Method:	
Route Loss of Connectivity Period[n]	array of seconds
Minimum Route Loss of Connectivity Period	seconds
Maximum Route Loss of Connectivity Period	seconds
Median Route Loss of Connectivity Period	seconds
Average Route Loss of Connectivity Period	seconds

8. Test Cases

It is RECOMMENDED that all applicable test cases be performed for best characterization of the DUT. The test cases follow a generic procedure tailored to the specific DUT configuration and Convergence Event [Pollt]. This generic procedure is as follows:

1. Establish DUT and Tester configurations and advertise an IGP topology from Tester to DUT.

2. Send Offered Load from Tester to DUT on ingress interface.
 3. Verify traffic is routed correctly. Verify if traffic is forwarded without Impaired Packets [Po06].
 4. Introduce Convergence Event [Pollt].
 5. Measure First Route Convergence Time [Pollt].
 6. Measure Full Convergence Time [Pollt].
 7. Stop Offered Load.
 8. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period [Pollt]. At the same time measure number of Impaired Packets [Pollt].
 9. Wait sufficient time for queues to drain. The duration of this time period MUST be larger than or equal to the Forwarding Delay Threshold.
 10. Restart Offered Load.
 11. Reverse Convergence Event.
 12. Measure First Route Convergence Time.
 13. Measure Full Convergence Time.
 14. Stop Offered Load.
 15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets [Pollt].
- 8.1. Interface Failure and Recovery
- 8.1.1. Convergence Due to Local Interface Failure and Recovery

Objective

To obtain the IGP convergence measurements for Local Interface failure and recovery events. The Next-Best Egress Interface can be a single interface (Figure 1) or an ECMP set (Figure 2). The test with ECMP topology (Figure 2) is OPTIONAL.

Procedure

1. Advertise an IGP topology from Tester to DUT using the topology shown in Figure 1 or 2.
 2. Send Offered Load from Tester to DUT on ingress interface.
 3. Verify traffic is forwarded over Preferred Egress Interface.
 4. Remove link on the Preferred Egress Interface of the DUT. This is the Convergence Event.
 5. Measure First Route Convergence Time.
 6. Measure Full Convergence Time.
 7. Stop Offered Load.
 8. Measure Route-Specific Convergence Times and Loss-Derived Convergence Time. At the same time measure number of Impaired Packets.
 9. Wait sufficient time for queues to drain.
 10. Restart Offered Load.
 11. Restore link on the Preferred Egress Interface of the DUT.
 12. Measure First Route Convergence Time.
 13. Measure Full Convergence Time.
 14. Stop Offered Load.
 15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.
- 8.1.2. Convergence Due to Remote Interface Failure and Recovery

Objective

To obtain the IGP convergence measurements for Remote Interface failure and recovery events. The Next-Best Egress Interface can be a single interface (Figure 3) or an ECMP set (Figure 4). The test with ECMP topology (Figure 4) is OPTIONAL.

Procedure

1. Advertise an IGP topology from Tester to SUT using the topology shown in Figure 3 or 4.
2. Send Offered Load from Tester to SUT on ingress interface.
3. Verify traffic is forwarded over Preferred Egress Interface.
4. Remove link on the interface of the Tester connected to the Preferred Egress Interface of the SUT. This is the Convergence Event.
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.
7. Stop Offered Load.
8. Measure Route-Specific Convergence Times and Loss-Derived Convergence Time. At the same time measure number of Impaired Packets.
9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. Restore link on the interface of the Tester connected to the Preferred Egress Interface of the SUT.
12. Measure First Route Convergence Time.
13. Measure Full Convergence Time.
14. Stop Offered Load.
15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

Discussion

In this test case there is a possibility of a packet forwarding loop that may occur transiently between DUT1 and DUT2 during convergence (micro-loop, see [Sh10]), which may increase the measured convergence times and loss of connectivity periods.

8.1.3. Convergence Due to ECMP Member Local Interface Failure and Recovery

Objective

To obtain the IGP convergence measurements for Local Interface link failure and recovery events of an ECMP Member.

Procedure

1. Advertise an IGP topology from Tester to DUT using the test setup shown in Figure 5.
2. Send Offered Load from Tester to DUT on ingress interface.
3. Verify traffic is forwarded over the ECMP member interface of the DUT that will be failed in the next step.
4. Remove link on one of the ECMP member interfaces of the DUT. This is the Convergence Event.
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.
7. Stop Offered Load.
8. Measure Route-Specific Convergence Times and Loss-Derived Convergence Time. At the same time measure number of Impaired Packets.
9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. Restore link on the ECMP member interface of the DUT.
12. Measure First Route Convergence Time.
13. Measure Full Convergence Time.
14. Stop Offered Load.
15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

8.1.4. Convergence Due to ECMP Member Remote Interface Failure and Recovery

Objective

To obtain the IGP convergence measurements for Remote Interface link failure and recovery events for an ECMP Member.

Procedure

1. Advertise an IGP topology from Tester to DUT using the test setup shown in Figure 6.
2. Send Offered Load from Tester to DUT on ingress interface.
3. Verify traffic is forwarded over the ECMP member interface of the DUT that will be failed in the next step.
4. Remove link on the interface of the Tester to R2. This is the Convergence Event Trigger.
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.
7. Stop Offered Load.
8. Measure Route-Specific Convergence Times and Loss-Derived Convergence Time. At the same time measure number of Impaired Packets.
9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. Restore link on the interface of the Tester to R2.
12. Measure First Route Convergence Time.
13. Measure Full Convergence Time.
14. Stop Offered Load.
15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

Discussion

In this test case there is a possibility of a packet forwarding loop that may occur temporarily between DUT1 and DUT2 during convergence (micro-loop, see [Sh10]), which may increase the measured convergence times and loss of connectivity periods.

8.1.5. Convergence Due to Parallel Link Interface Failure and Recovery

Objective

To obtain the IGP convergence measurements for local link failure and recovery events for a member of a parallel link. The links can be used for data load balancing

Procedure

1. Advertise an IGP topology from Tester to DUT using the test setup shown in Figure 7.
2. Send Offered Load from Tester to DUT on ingress interface.
3. Verify traffic is forwarded over the parallel link member that will be failed in the next step.
4. Remove link on one of the parallel link member interfaces of the DUT. This is the Convergence Event.
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.
7. Stop Offered Load.
8. Measure Route-Specific Convergence Times and Loss-Derived Convergence Time. At the same time measure number of Impaired Packets.
9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. Restore link on the Parallel Link member interface of the DUT.
12. Measure First Route Convergence Time.
13. Measure Full Convergence Time.

14. Stop Offered Load.
15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

8.2. Other Failures and Recoveries

8.2.1. Convergence Due to Layer 2 Session Loss and Recovery

Objective

To obtain the IGP convergence measurements for a local layer 2 loss and recovery.

Procedure

1. Advertise an IGP topology from Tester to DUT using the topology shown in Figure 1.
2. Send Offered Load from Tester to DUT on ingress interface.
3. Verify traffic is routed over Preferred Egress Interface.
4. Remove Layer 2 session from Preferred Egress Interface of the DUT. This is the Convergence Event.
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.
7. Stop Offered Load.
8. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.
9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. Restore Layer 2 session on Preferred Egress Interface of the DUT.
12. Measure First Route Convergence Time.

13. Measure Full Convergence Time.
14. Stop Offered Load.
15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

Discussion

When removing the layer 2 session, the physical layer must stay up. Configure IGP timers such that the IGP adjacency does not time out before layer 2 failure is detected.

To measure convergence time, traffic SHOULD start dropping on the Preferred Egress Interface on the instant the layer 2 session is removed. Alternatively the Tester SHOULD record the time the instant layer 2 session is removed and traffic loss SHOULD only be measured on the Next-Best Egress Interface. For loss-derived benchmarks the time of the Start Traffic Instant SHOULD be recorded as well. See Section 4.1.

8.2.2. Convergence Due to Loss and Recovery of IGP Adjacency

Objective

To obtain the IGP convergence measurements for loss and recovery of an IGP Adjacency. The IGP adjacency is removed on the Tester by disabling processing of IGP routing protocol packets on the Tester.

Procedure

1. Advertise an IGP topology from Tester to DUT using the topology shown in Figure 1.
2. Send Offered Load from Tester to DUT on ingress interface.
3. Verify traffic is routed over Preferred Egress Interface.
4. Remove IGP adjacency from the Preferred Egress Interface while the layer 2 session MUST be maintained. This is the Convergence Event.
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.

7. Stop Offered Load.
8. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.
9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. Restore IGP session on Preferred Egress Interface of the DUT.
12. Measure First Route Convergence Time.
13. Measure Full Convergence Time.
14. Stop Offered Load.
15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

Discussion

Configure layer 2 such that layer 2 does not time out before IGP adjacency failure is detected.

To measure convergence time, traffic SHOULD start dropping on the Preferred Egress Interface on the instant the IGP adjacency is removed. Alternatively the Tester SHOULD record the time the instant the IGP adjacency is removed and traffic loss SHOULD only be measured on the Next-Best Egress Interface. For loss-derived benchmarks the time of the Start Traffic Instant SHOULD be recorded as well. See Section 4.1.

8.2.3. Convergence Due to Route Withdrawal and Re-advertisement

Objective

To obtain the IGP convergence measurements for route withdrawal and re-advertisement.

Procedure

1. Advertise an IGP topology from Tester to DUT using the topology shown in Figure 1. The routes that will be withdrawn MUST be a set of leaf routes advertised by at least two nodes in the emulated topology. The topology SHOULD be such that before the withdrawal the DUT prefers the leaf routes advertised by a node "nodeA" via the Preferred Egress Interface, and after the withdrawal the DUT prefers the leaf routes advertised by a node "nodeB" via the Next-Best Egress Interface.
2. Send Offered Load from Tester to DUT on Ingress Interface.
3. Verify traffic is routed over Preferred Egress Interface.
4. The Tester withdraws the set of IGP leaf routes from nodeA. This is the Convergence Event. The withdrawal update message SHOULD be a single unfragmented packet. If the routes cannot be withdrawn by a single packet, the messages SHOULD be sent using the same pacing characteristics as the DUT. The Tester MAY record the time it sends the withdrawal message(s).
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.
7. Stop Offered Load.
8. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.
9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. Re-advertise the set of withdrawn IGP leaf routes from nodeA emulated by the Tester. The update message SHOULD be a single unfragmented packet. If the routes cannot be advertised by a single packet, the messages SHOULD be sent using the same pacing characteristics as the DUT. The Tester MAY record the time it sends the update message(s).
12. Measure First Route Convergence Time.
13. Measure Full Convergence Time.
14. Stop Offered Load.

15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

Discussion

To measure convergence time, traffic SHOULD start dropping on the Preferred Egress Interface on the instant the routes are withdrawn by the Tester. Alternatively the Tester SHOULD record the time the instant the routes are withdrawn and traffic loss SHOULD only be measured on the Next-Best Egress Interface. For loss-derived benchmarks the time of the Start Traffic Instant SHOULD be recorded as well. See Section 4.1.

8.3. Administrative changes

8.3.1. Convergence Due to Local Interface Administrative Changes

Objective

To obtain the IGP convergence measurements for administratively disabling and enabling a Local Interface.

Procedure

1. Advertise an IGP topology from Tester to DUT using the topology shown in Figure 1.
2. Send Offered Load from Tester to DUT on ingress interface.
3. Verify traffic is routed over Preferred Egress Interface.
4. Administratively disable the Preferred Egress Interface of the DUT. This is the Convergence Event.
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.
7. Stop Offered Load.
8. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. Administratively enable the Preferred Egress Interface of the DUT..
12. Measure First Route Convergence Time.
13. Measure Full Convergence Time.
14. Stop Offered Load.
15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

8.3.2. Convergence Due to Cost Change

Objective

To obtain the IGP convergence measurements for route cost change.

Procedure

1. Advertise an IGP topology from Tester to DUT using the topology shown in Figure 1.
2. Send Offered Load from Tester to DUT on ingress interface.
3. Verify traffic is routed over Preferred Egress Interface.
4. The Tester, emulating the neighbor node, increases the cost for all IGP routes at Preferred Egress Interface of the DUT so that the Next-Best Egress Interface becomes preferred path. The update message advertising the higher cost MUST be a single unfragmented packet. This is the Convergence Event. The Tester MAY record the time it sends the update message advertising the higher cost on the Preferred Egress Interface.
5. Measure First Route Convergence Time.
6. Measure Full Convergence Time.
7. Stop Offered Load.

8. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.
9. Wait sufficient time for queues to drain.
10. Restart Offered Load.
11. The Tester, emulating the neighbor node, decreases the cost for all IGP routes at Preferred Egress Interface of the DUT so that the Preferred Egress Interface becomes preferred path. The update message advertising the lower cost MUST be a single unfragmented packet.
12. Measure First Route Convergence Time.
13. Measure Full Convergence Time.
14. Stop Offered Load.
15. Measure Route-Specific Convergence Times, Loss-Derived Convergence Time, Route Loss of Connectivity Periods, and Loss-Derived Loss of Connectivity Period. At the same time measure number of Impaired Packets.

Discussion

To measure convergence time, traffic SHOULD start dropping on the Preferred Egress Interface on the instant the cost is changed by the Tester. Alternatively the Tester SHOULD record the time the instant the cost is changed and traffic loss SHOULD only be measured on the Next-Best Egress Interface. For loss-derived benchmarks the time of the Start Traffic Instant SHOULD be recorded as well. See Section 4.1.

9. Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the constraints specified in the sections above.

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT/SUT.

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

10. IANA Considerations

This document requires no IANA considerations.

11. Acknowledgements

Thanks to Sue Hares, Al Morton, Kevin Dubray, Ron Bonica, David Ward, Peter De Vriendt, Anuj Dewagan, Julien Meuric, Adrian Farrel, Stewart Bryant, and the Benchmarking Methodology Working Group for their contributions to this work.

12. References

12.1. Normative References

- [Br91] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [Br97] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [Br99] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [Ca90] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, December 1990.
- [Co08] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, July 2008.
- [De02] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, November 2002.
- [Ho08] Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, October 2008.

- [Ko02] Koodli, R. and R. Ravikanth, "One-way Loss Pattern Sample Metrics", RFC 3357, August 2002.
- [Ma05] Manral, V., White, R., and A. Shaikh, "Benchmarking Basic OSPF Single Router Control Plane Convergence", RFC 4061, April 2005.
- [Ma05c] Manral, V., White, R., and A. Shaikh, "Considerations When Using Basic OSPF Convergence Benchmarks", RFC 4063, April 2005.
- [Ma05t] Manral, V., White, R., and A. Shaikh, "OSPF Benchmarking Terminology and Concepts", RFC 4062, April 2005.
- [Ma98] Mandeville, R., "Benchmarking Terminology for LAN Switching Devices", RFC 2285, February 1998.
- [Mo98] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.
- [Ne07] Newman, D. and T. Player, "Hash and Stuffing: Overlooked Factors in Network Device Benchmarking", RFC 4814, March 2007.
- [Pa05] Pan, P., Swallow, G., and A. Atlas, "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090, May 2005.
- [Po06] Poretsky, S., Perser, J., Erramilli, S., and S. Khurana, "Terminology for Benchmarking Network-layer Traffic Control Mechanisms", RFC 4689, October 2006.
- [Pol1t] Poretsky, S., Imhoff, B., and K. Michielsen, "Terminology for Benchmarking Link-State IGP Data Plane Route Convergence", draft-ietf-bmwg-igp-dataplane-conv-term-23 (work in progress), January 2011.
- [Sh10] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, January 2010.
- [Sh10i] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, January 2010.
- [Th00] Thaler, D. and C. Hopps, "Multipath Issues in Unicast and Multicast Next-Hop Selection", RFC 2991, November 2000.

12.2. Informative References

- [Al00] Alaettinoglu, C., Jacobson, V., and H. Yu, "Towards Millisecond IGP Convergence", NANOG 20, October 2000.

- [Al02] Alaettinoglu, C. and S. Casner, "ISIS Routing on the Qwest Backbone: a Recipe for Subsecond ISIS Convergence", NANOG 24, February 2002.
- [Fi02] Filsfils, C., "Tutorial: Deploying Tight-SLA Services on an Internet Backbone: ISIS Fast Convergence and Differentiated Services Design", NANOG 25, June 2002.
- [Fr05] Francois, P., Filsfils, C., Evans, J., and O. Bonaventure, "Achieving SubSecond IGP Convergence in Large IP Networks", ACM SIGCOMM Computer Communication Review v.35 n.3, July 2005.
- [Ka02] Katz, D., "Why are we scared of SPF? IGP Scaling and Stability", NANOG 25, June 2002.
- [Vi02] Villamizar, C., "Convergence and Restoration Techniques for ISP Interior Routing", NANOG 25, June 2002.

Authors' Addresses

Scott Poretsky
Allot Communications
67 South Bedford Street, Suite 400
Burlington, MA 01803
USA

Phone: + 1 508 309 2179
Email: sporetsky@allot.com

Brent Imhoff
Juniper Networks
1194 North Mathilda Ave
Sunnyvale, CA 94089
USA

Phone: + 1 314 378 2571
Email: bimhoff@planetisp.com

Kris Michiels
Cisco Systems
6A De Kleetlaan
Diegem, BRABANT 1831
Belgium

Email: kmichiel@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 13, 2011

S. Poretsky
Allot Communications
B. Imhoff
Juniper Networks
K. Michielsen
Cisco Systems
February 16, 2011

Terminology for Benchmarking Link-State IGP Data Plane Route Convergence
draft-ietf-bmwg-igp-dataplane-conv-term-23

Abstract

This document describes the terminology for benchmarking link-state Interior Gateway Protocol (IGP) route convergence. The terminology is to be used for benchmarking IGP convergence time through externally observable (black box) data plane measurements. The terminology can be applied to any link-state IGP, such as Intermediate System to Intermediate System (IS-IS) and Open Shortest Path First (OSPF).

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 <http://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 August 13, 2011.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1.	Introduction and Scope	4
2.	Existing Definitions	4
3.	Term Definitions	4
3.1.	Convergence Types	4
3.1.1.	Route Convergence	5
3.1.2.	Full Convergence	5
3.2.	Instants	6
3.2.1.	Traffic Start Instant	6
3.2.2.	Convergence Event Instant	6
3.2.3.	Convergence Recovery Instant	7
3.2.4.	First Route Convergence Instant	7
3.3.	Transitions	8
3.3.1.	Convergence Event Transition	8
3.3.2.	Convergence Recovery Transition	9
3.4.	Interfaces	9
3.4.1.	Local Interface	9
3.4.2.	Remote Interface	9
3.4.3.	Preferred Egress Interface	10
3.4.4.	Next-Best Egress Interface	10
3.5.	Benchmarking Methods	11
3.5.1.	Rate-Derived Method	11
3.5.2.	Loss-Derived Method	13
3.5.3.	Route-Specific Loss-Derived Method	14
3.6.	Benchmarks	15
3.6.1.	Full Convergence Time	15
3.6.2.	First Route Convergence Time	16
3.6.3.	Route-Specific Convergence Time	17
3.6.4.	Loss-Derived Convergence Time	18
3.6.5.	Route Loss of Connectivity Period	19
3.6.6.	Loss-Derived Loss of Connectivity Period	20
3.7.	Measurement Terms	21
3.7.1.	Convergence Event	21
3.7.2.	Convergence Packet Loss	21
3.7.3.	Connectivity Packet Loss	22
3.7.4.	Packet Sampling Interval	22
3.7.5.	Sustained Convergence Validation Time	23
3.7.6.	Forwarding Delay Threshold	24
3.8.	Miscellaneous Terms	24
3.8.1.	Impaired Packet	24
4.	Security Considerations	24
5.	IANA Considerations	25
6.	Acknowledgements	25
7.	Normative References	25
	Authors' Addresses	26

1. Introduction and Scope

This document is a companion to [Pollm] which the methodology to be used for benchmarking link-state Interior Gateway Protocol (IGP) Convergence by observing the data plane. The purpose of this document is to introduce new terms required to complete execution of the Link-State IGP Data Plane Route Convergence methodology [Pollm].

IGP convergence time is measured by observing the dataplane through the Device Under Test (DUT) at the Tester. The methodology and terminology to be used for benchmarking IGP Convergence can be applied to IPv4 and IPv6 traffic and link-state IGPs such as Intermediate System to Intermediate System (IS-IS) [Ca90][Ho08], Open Shortest Path First (OSPF) [Mo98][Co08], and others.

2. Existing Definitions

This document uses existing terminology defined in other IETF documents. Examples include, but are not limited to:

Throughput	[Ref.[Br91], section 3.17]
Offered Load	[Ref.[Ma98], section 3.5.2]
Forwarding Rate	[Ref.[Ma98], section 3.6.1]
Device Under Test (DUT)	[Ref.[Ma98], section 3.1.1]
System Under Test (SUT)	[Ref.[Ma98], section 3.1.2]
Out-of-Order Packet	[Ref.[Po06], section 3.3.4]
Duplicate Packet	[Ref.[Po06], section 3.3.5]
Stream	[Ref.[Po06], section 3.3.2]
Forwarding Delay	[Ref.[Po06], section 3.2.4]
IP Packet Delay Variation (IPDV)	[Ref.[De02], section 1.2]
Loss Period	[Ref.[Ko02], section 4]

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 BCP 14, RFC 2119 [Br97]. RFC 2119 defines the use of these key words to help make the intent of standards track documents as clear as possible. While this document uses these keywords, this document is not a standards track document.

3. Term Definitions

3.1. Convergence Types

3.1.1.1. Route Convergence

Definition:

The process of updating all components of the router, including the Routing Information Base (RIB) and Forwarding Information Base (FIB), along with software and hardware tables, with the most recent route change(s) such that forwarding for a route entry is successful on the Next-Best Egress Interface [Section 3.4.4].

Discussion:

In general IGP convergence does not necessarily result in a change in forwarding. But the test cases in [Pollm] are specified such that the IGP convergence results in a change of egress interface for the measurement dataplane traffic. Due to this property of the test case specifications, Route Convergence can be observed externally by the rerouting of the measurement dataplane traffic to the Next-best Egress Interface [Section 3.4.4].

Measurement Units: N/A

See Also:

Next-Best Egress Interface, Full Convergence

3.1.1.2. Full Convergence

Definition:

Route Convergence for all routes in the Forwarding Information Base (FIB).

Discussion:

In general IGP convergence does not necessarily result in a change in forwarding. But the test cases in [Pollm] are specified such that the IGP convergence results in a change of egress interface for the measurement dataplane traffic. Due to this property of the test cases specifications, Full Convergence can be observed externally by the rerouting of the measurement dataplane traffic to the Next-best Egress Interface [Section 3.4.4].

Measurement Units: N/A

See Also:

Next-Best Egress Interface, Route Convergence

3.2. Instants

3.2.1. Traffic Start Instant

Definition:

The time instant the Tester sends out the first data packet to the Device Under Test (DUT).

Discussion:

If using the Loss-Derived Method [Section 3.5.2] or the Route-Specific Loss-Derived Method [Section 3.5.3] to benchmark IGP convergence time, and the applied Convergence Event [Section 3.7.1] does not cause instantaneous traffic loss for all routes at the Convergence Event Instant [Section 3.2.2] then the Tester SHOULD collect a timestamp on the Traffic Start Instant in order to measure the period of time between the Traffic Start Instant and Convergence Event Instant.

Measurement Units:

seconds (and fractions), reported with resolution sufficient to distinguish between different instants

See Also:

Loss-Derived Method, Route-Specific Loss-Derived Method, Convergence Event, Convergence Event Instant

3.2.2. Convergence Event Instant

Definition:

The time instant that a Convergence Event [Section 3.7.1] occurs.

Discussion:

If the Convergence Event [Section 3.7.1] causes instantaneous traffic loss on the Preferred Egress Interface [Section 3.4.3], the Convergence Event Instant is observable from the data plane as the instant that no more packets are received on the Preferred Egress Interface.

The Tester SHOULD collect a timestamp on the Convergence Event Instant if it the Convergence Event does not cause instantaneous traffic loss on the Preferred Egress Interface [Section 3.4.3].

Measurement Units:

seconds (and fractions), reported with resolution sufficient to distinguish between different instants

See Also:

Convergence Event, Preferred Egress Interface

3.2.3. Convergence Recovery Instant

Definition:

The time instant that Full Convergence [Section 3.1.2] has completed.

Discussion:

The Full Convergence completed state MUST be maintained for an interval of duration equal to the Sustained Convergence Validation Time [Section 3.7.5] in order to validate the Convergence Recovery Instant.

The Convergence Recovery Instant is observable from the data plane as the instant the Device Under Test (DUT) forwards traffic to all destinations over the Next-Best Egress Interface [Section 3.4.4] without impairments.

Measurement Units:

seconds (and fractions), reported with resolution sufficient to distinguish between different instants

See Also:

Sustained Convergence Validation Time, Full Convergence, Next-Best Egress Interface

3.2.4. First Route Convergence Instant

Definition:

The time instant the first route entry completes Route Convergence [Section 3.1.1]

Discussion:

Any route may be the first to complete Route Convergence. The First Route Convergence Instant is observable from the data plane as the

instant that the first packet that is not an Impaired Packet [Section 3.8.1] is received from the Next-Best Egress Interface [Section 3.4.4] or, for the test cases with Equal Cost Multi-Path (ECMP) or Parallel Links, the instant that the Forwarding Rate on the Next-Best Egress Interface [Section 3.4.4] starts to increase.

Measurement Units:

seconds (and fractions), reported with resolution sufficient to distinguish between different instants

See Also:

Route Convergence, Impaired Packet, Next-Best Egress Interface

3.3. Transitions

3.3.1. Convergence Event Transition

Definition:

A time interval following a Convergence Event [Section 3.7.1] in which Forwarding Rate on the Preferred Egress Interface [Section 3.4.3] gradually reduces to zero.

Discussion:

The Forwarding Rate during a Convergence Event Transition may or may not decrease linearly.

The Forwarding Rate observed on the Device Under Test (DUT) egress interface(s) may or may not decrease to zero.

The Offered Load, the number of routes, and the Packet Sampling Interval [Section 3.7.4] influence the observations of the Convergence Event Transition using the Rate-Derived Method [Section 3.5.1].

Measurement Units: seconds (and fractions)

See Also:

Convergence Event, Preferred Egress Interface, Packet Sampling Interval, Rate-Derived Method

3.3.2. Convergence Recovery Transition

Definition:

A time interval following the First Route Convergence Instant [Section 3.4.4] in which Forwarding Rate on the Device Under Test (DUT) egress interface(s) gradually increases to equal the Offered Load.

Discussion:

The Forwarding Rate observed during a Convergence Recovery Transition may or may not increase linearly.

The Offered Load, the number of routes, and the Packet Sampling Interval [Section 3.7.4] influence the observations of the Convergence Recovery Transition using the Rate-Derived Method [Section 3.5.1].

Measurement Units: seconds (and fractions)

See Also:

First Route Convergence Instant, Packet Sampling Interval, Rate-Derived Method

3.4. Interfaces

3.4.1. Local Interface

Definition:

An interface on the Device Under Test (DUT).

Discussion:

A failure of a Local Interface indicates that the failure occurred directly on the Device Under Test (DUT).

Measurement Units: N/A

See Also: Remote Interface

3.4.2. Remote Interface

Definition:

An interface on a neighboring router that is not directly connected

to any interface on the Device Under Test (DUT).

Discussion:

A failure of a Remote Interface indicates that the failure occurred on a neighbor router's interface that is not directly connected to the Device Under Test (DUT).

Measurement Units: N/A

See Also: Local Interface

3.4.3. Preferred Egress Interface

Definition:

The outbound interface from the Device Under Test (DUT) for traffic routed to the preferred next-hop.

Discussion:

The Preferred Egress Interface is the egress interface prior to a Convergence Event [Section 3.7.1].

Measurement Units: N/A

See Also: Convergence Event, Next-Best Egress Interface

3.4.4. Next-Best Egress Interface

Definition:

The outbound interface or set of outbound interfaces in an Equal Cost Multipath (ECMP) set or parallel link set of the Device Under Test (DUT) for traffic routed to the second-best next-hop.

Discussion:

The Next-Best Egress Interface becomes the egress interface after a Convergence Event [Section 3.4.4].

For the test cases in [Pollm] using test topologies with an ECMP set or parallel link set, the term Preferred Egress Interface refers to all members of the link set.

Measurement Units: N/A

See Also: Convergence Event, Preferred Egress Interface

3.5. Benchmarking Methods

3.5.1. Rate-Derived Method

Definition:

The method to calculate convergence time benchmarks from observing Forwarding Rate each Packet Sampling Interval [Section 3.7.4].

Discussion:

Figure 1 shows an example of the Forwarding Rate change in time during convergence as observed when using the Rate-Derived Method.

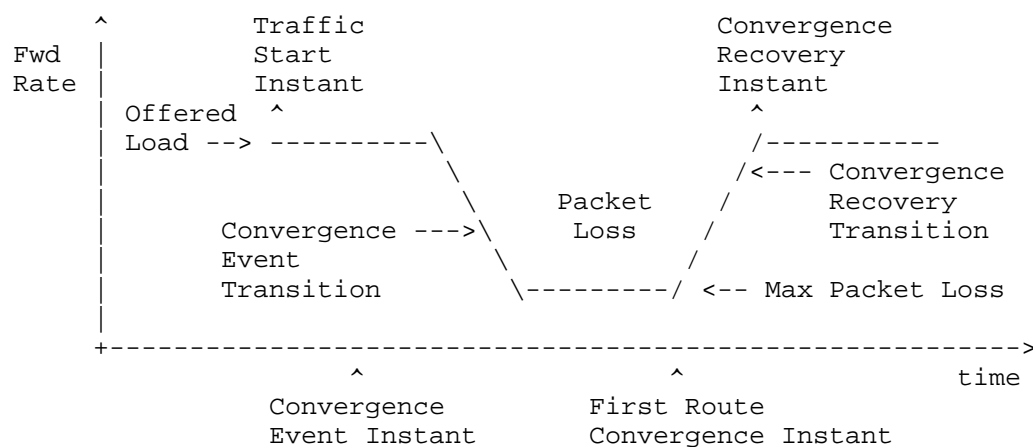


Figure 1: Rate-Derived Convergence Graph

To enable collecting statistics of Out-of-Order Packets per flow (See [Th00], Section 3) the Offered Load SHOULD consist of multiple Streams [Po06] and each Stream SHOULD consist of a single flow . If sending multiple Streams, the measured traffic statistics for all Streams MUST be added together.

The destination addresses for the Offered Load MUST be distributed such that all routes or a statistically representative subset of all routes are matched and each of these routes is offered an equal share of the Offered Load. It is RECOMMENDED to send traffic to all routes, but a statistically representative subset of all routes can be used if required.

At least one packet per route for all routes matched in the Offered

Load MUST be offered to the DUT within each Packet Sampling Interval. For maximum accuracy the value for the Packet Sampling Interval SHOULD be as small as possible, but the presence of IP Packet Delay Variation (IPDV) [De02] may enforce using a larger Packet Sampling Interval.

The Offered Load, IPDV, the number of routes, and the Packet Sampling Interval influence the observations for the Rate-Derived Method. It may be difficult to identify the different convergence time instants in the Rate-Derived Convergence Graph. For example, it is possible that a Convergence Event causes the Forwarding Rate to drop to zero, while this may not be observed in the Forwarding Rate measurements if the Packet Sampling Interval is too large.

IPDV causes fluctuations in the number of received packets during each Packet Sampling Interval. To account for the presence of IPDV in determining if a convergence instant has been reached, Forwarding Delay SHOULD be observed during each Packet Sampling Interval. The minimum and maximum number of packets expected in a Packet Sampling Interval in presence of IPDV can be calculated with Equation 1.

number of packets expected in a Packet Sampling Interval
in presence of IP Packet Delay Variation
= expected number of packets without IP Packet Delay Variation
+/- ((maxDelay - minDelay) * Offered Load)
with minDelay and maxDelay the minimum resp. maximum Forwarding Delay
of packets received during the Packet Sampling Interval

Equation 1

To determine if a convergence instant has been reached the number of packets received in a Packet Sampling Interval is compared with the range of expected number of packets calculated in Equation 1.

If packets are going over multiple ECMP members and one or more of the members has failed then the number of received packets during each Packet Sampling Interval may vary, even excluding presence of IPDV. To prevent fluctuation of the number of received packets during each Packet Sampling Interval for this reason, the Packet Sampling Interval duration SHOULD be a whole multiple of the time between two consecutive packets sent to the same destination.

Metrics measured at the Packet Sampling Interval MUST include Forwarding Rate and Impaired Packet count.

To measure convergence time benchmarks for Convergence Events [Section 3.7.1] that do not cause instantaneous traffic loss for all routes at the Convergence Event Instant, the Tester SHOULD collect a

timestamp of the Convergence Event Instant [Section 3.2.2] and the Tester SHOULD observe Forwarding Rate separately on the Next-Best Egress Interface.

Since the Rate-Derived Method does not distinguish between individual traffic destinations, it SHOULD NOT be used for any route specific measurements. Therefor Rate-Derived Method SHOULD NOT be used to benchmark Route Loss of Connectivity Period [Section 3.6.5].

Measurement Units: N/A

See Also:

Packet Sampling Interval, Convergence Event, Convergence Event Instant, Next-Best Egress Interface, Route Loss of Connectivity Period

3.5.2. Loss-Derived Method

Definition:

The method to calculate the Loss-Derived Convergence Time [Section 3.6.4] and Loss-Derived Loss of Connectivity Period [Section 3.6.6] benchmarks from the amount of Impaired Packets [Section 3.8.1].

Discussion:

To enable collecting statistics of Out-of-Order Packets per flow (See [Th00], Section 3) the Offered Load SHOULD consist of multiple Streams [Po06] and each Stream SHOULD consist of a single flow . If sending multiple Streams, the measured traffic statistics for all Streams MUST be added together.

The destination addresses for the Offered Load MUST be distributed such that all routes or a statistically representative subset of all routes are matched and each of these routes is offered an equal share of the Offered Load. It is RECOMMENDED to send traffic to all routes, but a statistically representative subset of all routes can be used if required.

Loss-Derived Method SHOULD always be combined with Rate-Derived Method in order to observe Full Convergence completion. The total amount of Convergence Packet Loss is collected after Full Convergence completion.

To measure convergence time and loss of connectivity benchmarks for Convergence Events that cause instantaneous traffic loss for all

routes at the Convergence Event Instant, the Tester SHOULD observe Impaired Packet count on all DUT egress interfaces (see Connectivity Packet Loss [Section 3.7.3]).

To measure convergence time benchmarks for Convergence Events that do not cause instantaneous traffic loss for all routes at the Convergence Event Instant, the Tester SHOULD collect timestamps of the Start Traffic Instant and of the Convergence Event Instant, and the Tester SHOULD observe Impaired Packet count separately on the Next-Best Egress Interface (See Convergence Packet Loss [Section 3.7.2]).

Since Loss-Derived Method does not distinguish between traffic destinations and the Impaired Packet statistics are only collected after Full Convergence completion, this method can only be used to measure average values over all routes. For these reasons Loss-Derived Method can only be used to benchmark Loss-Derived Convergence Time [Section 3.6.4] and Loss-Derived Loss of Connectivity Period [Section 3.6.6].

Note that the Loss-Derived Method measures an average over all routes, including the routes that may not be impacted by the Convergence Event, such as routes via non-impacted members of ECMP or parallel links.

Measurement Units: N/A

See Also:

Loss-Derived Convergence Time, Loss-Derived Loss of Connectivity Period, Connectivity Packet Loss, Convergence Packet Loss

3.5.3. Route-Specific Loss-Derived Method

Definition:

The method to calculate the Route-Specific Convergence Time [Section 3.6.3] benchmark from the amount of Impaired Packets [Section 3.8.1] during convergence for a specific route entry.

Discussion:

To benchmark Route-Specific Convergence Time, the Tester provides an Offered Load that consists of multiple Streams [Po06]. Each Stream has a single destination address matching a different route entry, for all routes or a statistically representative subset of all routes. Each Stream SHOULD consist of a single flow (See [Th00], Section 3). Convergence Packet Loss is measured for each Stream

separately.

Route-Specific Loss-Derived Method SHOULD always be combined with Rate-Derived Method in order to observe Full Convergence completion. The total amount of Convergence Packet Loss [Section 3.7.2] for each Stream is collected after Full Convergence completion.

Route-Specific Loss-Derived Method is the RECOMMENDED method to measure convergence time benchmarks.

To measure convergence time and loss of connectivity benchmarks for Convergence Events that cause instantaneous traffic loss for all routes at the Convergence Event Instant, the Tester SHOULD observe Impaired Packet count on all DUT egress interfaces (see Connectivity Packet Loss [Section 3.7.3]).

To measure convergence time benchmarks for Convergence Events that do not cause instantaneous traffic loss for all routes at the Convergence Event Instant, the Tester SHOULD collect timestamps of the Start Traffic Instant and of the Convergence Event Instant, and the Tester SHOULD observe packet loss separately on the Next-Best Egress Interface (See Convergence Packet Loss [Section 3.7.2]).

Since Route-Specific Loss-Derived Method uses traffic streams to individual routes, it observes Impaired Packet count as it would be experienced by a network user. For this reason Route-Specific Loss-Derived Method is RECOMMENDED to measure Route-Specific Convergence Time benchmarks and Route Loss of Connectivity Period benchmarks.

Measurement Units: N/A

See Also:

Route-Specific Convergence Time, Route Loss of Connectivity Period, Connectivity Packet Loss, Convergence Packet Loss

3.6. Benchmarks

3.6.1. Full Convergence Time

Definition:

The time duration of the period between the Convergence Event Instant and the Convergence Recovery Instant as observed using the Rate-Derived Method.

Discussion:

Using the Rate-Derived Method, Full Convergence Time can be calculated as the time difference between the Convergence Event Instant and the Convergence Recovery Instant, as shown in Equation 2.

$$\text{Full Convergence Time} = \text{Convergence Recovery Instant} - \text{Convergence Event Instant}$$

Equation 2

The Convergence Event Instant can be derived from the Forwarding Rate observation or from a timestamp collected by the Tester.

For the test cases described in [Pollm], it is expected that Full Convergence Time equals the maximum Route-Specific Convergence Time when benchmarking all routes in FIB using the Route-Specific Loss-Derived Method.

It is not possible to measure Full Convergence Time using the Loss-Derived Method.

Measurement Units: seconds (and fractions)

See Also:

Full Convergence, Rate-Derived Method, Route-Specific Loss-Derived Method, Convergence Event Instant, Convergence Recovery Instant

3.6.2. First Route Convergence Time

Definition:

The duration of the period between the Convergence Event Instant and the First Route Convergence Instant as observed using the Rate-Derived Method.

Discussion:

Using the Rate-Derived Method, First Route Convergence Time can be calculated as the time difference between the Convergence Event Instant and the First Route Convergence Instant, as shown with Equation 3.

$$\text{First Route Convergence Time} = \text{First Route Convergence Instant} - \text{Convergence Event Instant}$$

Equation 3

The Convergence Event Instant can be derived from the Forwarding Rate

observation or from a timestamp collected by the Tester.

For the test cases described in [Pollm], it is expected that First Route Convergence Time equals the minimum Route-Specific Convergence Time when benchmarking all routes in FIB using the Route-Specific Loss-Derived Method.

It is not possible to measure First Route Convergence Time using the Loss-Derived Method.

Measurement Units: seconds (and fractions)

See Also:

Rate-Derived Method, Route-Specific Loss-Derived Method, Convergence Event Instant, First Route Convergence Instant

3.6.3. Route-Specific Convergence Time

Definition:

The amount of time it takes for Route Convergence to be completed for a specific route, as calculated from the amount of Impaired Packets [Section 3.8.1] during convergence for a single route entry.

Discussion:

Route-Specific Convergence Time can only be measured using the Route-Specific Loss-Derived Method.

If the applied Convergence Event causes instantaneous traffic loss for all routes at the Convergence Event Instant, Connectivity Packet Loss should be observed. Connectivity Packet Loss is the combined Impaired Packet count observed on Preferred Egress Interface and Next-Best Egress Interface. When benchmarking Route-Specific Convergence Time, Connectivity Packet Loss is measured and Equation 4 is applied for each measured route. The calculation is equal to Equation 8 in Section 3.6.5.

Route-Specific Convergence Time =
Connectivity Packet Loss for specific route/Offered Load per route

Equation 4

If the applied Convergence Event does not cause instantaneous traffic loss for all routes at the Convergence Event Instant, then the Tester SHOULD collect timestamps of the Traffic Start Instant and of the Convergence Event Instant, and the Tester SHOULD observe Convergence

Packet Loss separately on the Next-Best Egress Interface. When benchmarking Route-Specific Convergence Time, Convergence Packet Loss is measured and Equation 5 is applied for each measured route.

Route-Specific Convergence Time =
Convergence Packet Loss for specific route/Offered Load per route
- (Convergence Event Instant - Traffic Start Instant)

Equation 5

The Route-Specific Convergence Time benchmarks enable minimum, maximum, average, and median convergence time measurements to be reported by comparing the results for the different route entries. It also enables benchmarking of convergence time when configuring a priority value for route entry(ies). Since multiple Route-Specific Convergence Times can be measured it is possible to have an array of results. The format for reporting Route-Specific Convergence Time is provided in [Pollm].

Measurement Units: seconds (and fractions)

See Also:

Route-Specific Loss-Derived Method, Convergence Event, Convergence Event Instant, Convergence Packet Loss, Connectivity Packet Loss, Route Convergence

3.6.4. Loss-Derived Convergence Time

Definition:

The average Route Convergence time for all routes in the Forwarding Information Base (FIB), as calculated from the amount of Impaired Packets [Section 3.8.1] during convergence.

Discussion:

Loss-Derived Convergence Time is measured using the Loss-Derived Method.

If the applied Convergence Event causes instantaneous traffic loss for all routes at the Convergence Event Instant, Connectivity Packet Loss [Section 3.7.3] should be observed. Connectivity Packet Loss is the combined Impaired Packet count observed on Preferred Egress Interface and Next-Best Egress Interface. When benchmarking Loss-Derived Convergence Time, Connectivity Packet Loss is measured and Equation 6 is applied.

$$\text{Loss-Derived Convergence Time} = \frac{\text{Connectivity Packet Loss}}{\text{Offered Load}}$$

Equation 6

If the applied Convergence Event does not cause instantaneous traffic loss for all routes at the Convergence Event Instant, then the Tester SHOULD collect timestamps of the Start Traffic Instant and of the Convergence Event Instant and the Tester SHOULD observe Convergence Packet Loss [Section 3.7.2] separately on the Next-Best Egress Interface. When benchmarking Loss-Derived Convergence Time, Convergence Packet Loss is measured and Equation 7 is applied.

$$\begin{aligned} \text{Loss-Derived Convergence Time} = & \frac{\text{Convergence Packet Loss}}{\text{Offered Load}} \\ & - (\text{Convergence Event Instant} - \text{Traffic Start Instant}) \end{aligned}$$

Equation 7

Measurement Units: seconds (and fractions)

See Also:

Convergence Packet Loss, Connectivity Packet Loss, Route Convergence, Loss-Derived Method

3.6.5. Route Loss of Connectivity Period

Definition:

The time duration of packet impairments for a specific route entry following a Convergence Event until Full Convergence completion, as observed using the Route-Specific Loss-Derived Method.

Discussion:

In general the Route Loss of Connectivity Period is not equal to the Route-Specific Convergence Time. If the DUT continues to forward traffic to the Preferred Egress Interface after the Convergence Event is applied then the Route Loss of Connectivity Period will be smaller than the Route-Specific Convergence Time. This is also specifically the case after reversing a failure event.

The Route Loss of Connectivity Period may be equal to the Route-Specific Convergence Time if, as a characteristic of the Convergence Event, traffic for all routes starts dropping instantaneously on the Convergence Event Instant. See discussion in [Pollm].

For the test cases described in [Pollm] the Route Loss of Connectivity Period is expected to be a single Loss Period [Ko02].

When benchmarking Route Loss of Connectivity Period, Connectivity Packet Loss is measured for each route and Equation 8 is applied for each measured route entry. The calculation is equal to Equation 4 in Section 3.6.3.

Route Loss of Connectivity Period =
Connectivity Packet Loss for specific route/Offered Load per route

Equation 8

Route Loss of Connectivity Period SHOULD be measured using Route-Specific Loss-Derived Method.

Measurement Units: seconds (and fractions)

See Also:

Route-Specific Convergence Time, Route-Specific Loss-Derived Method, Connectivity Packet Loss

3.6.6. Loss-Derived Loss of Connectivity Period

Definition:

The average time duration of packet impairments for all routes following a Convergence Event until Full Convergence completion, as observed using the Loss-Derived Method.

Discussion:

In general the Loss-Derived Loss of Connectivity Period is not equal to the Loss-Derived Convergence Time. If the DUT continues to forward traffic to the Preferred Egress Interface after the Convergence Event is applied then the Loss-Derived Loss of Connectivity Period will be smaller than the Loss-Derived Convergence Time. This is also specifically the case after reversing a failure event.

The Loss-Derived Loss of Connectivity Period may be equal to the Loss-Derived Convergence Time if, as a characteristic of the Convergence Event, traffic for all routes starts dropping instantaneously on the Convergence Event Instant. See discussion in [Pollm].

For the test cases described in [Pollm] each route's Route Loss of

Connectivity Period is expected to be a single Loss Period [Ko02].

When benchmarking Loss-Derived Loss of Connectivity Period, Connectivity Packet Loss is measured for all routes and Equation 9 is applied. The calculation is equal to Equation 6 in Section 3.6.4.

$$\text{Loss-Derived Loss of Connectivity Period} = \frac{\text{Connectivity Packet Loss for all routes}}{\text{Offered Load}}$$

Equation 9

Loss-Derived Loss of Connectivity Period SHOULD be measured using Loss-Derived Method.

Measurement Units: seconds (and fractions)

See Also:

Loss-Derived Convergence Time, Loss-Derived Method, Connectivity Packet Loss

3.7. Measurement Terms

3.7.1. Convergence Event

Definition:

The occurrence of an event in the network that will result in a change in the egress interface of the Device Under Test (DUT) for routed packets.

Discussion:

All test cases in [Pollm] are defined such that a Convergence Event results in a change of egress interface of the DUT. Local or remote triggers that cause a route calculation which does not result in a change in forwarding are not considered.

Measurement Units: N/A

See Also: Convergence Event Instant

3.7.2. Convergence Packet Loss

Definition:

The number of Impaired Packets [Section 3.8.1] as observed on the Next-Best Egress Interface of the DUT during convergence.

Discussion:

An Impaired Packet is considered as a lost packet.

Measurement Units: number of packets

See Also:

Connectivity Packet Loss

3.7.3. Connectivity Packet Loss

Definition:

The number of Impaired Packets observed on all DUT egress interfaces during convergence.

Discussion:

An Impaired Packet is considered as a lost packet. Connectivity Packet Loss is equal to Convergence Packet Loss if the Convergence Event causes instantaneous traffic loss for all egress interfaces of the DUT except for the Next-Best Egress Interface.

Measurement Units: number of packets

See Also:

Convergence Packet Loss

3.7.4. Packet Sampling Interval

Definition:

The interval at which the Tester (test equipment) polls to make measurements for arriving packets.

Discussion:

At least one packet per route for all routes matched in the Offered Load MUST be offered to the DUT within the Packet Sampling Interval. Metrics measured at the Packet Sampling Interval MUST include Forwarding Rate and received packets.

Packet Sampling Interval can influence the convergence graph as observed with the Rate-Derived Method. This is particularly true when implementations complete Full Convergence in less time than the Packet Sampling Interval. The Convergence Event Instant and First

Route Convergence Instant may not be easily identifiable and the Rate-Derived Method may produce a larger than actual convergence time.

Using a small Packet Sampling Interval in the presence of IPDV [De02] may cause fluctuations of the Forwarding Rate observation and can prevent correct observation of the different convergence time instants.

The value of the Packet Sampling Interval only contributes to the measurement accuracy of the Rate-Derived Method. For maximum accuracy the value for the Packet Sampling Interval SHOULD be as small as possible, but the presence of IPDV may enforce using a larger Packet Sampling Interval.

Measurement Units: seconds (and fractions)

See Also: Rate-Derived Method

3.7.5. Sustained Convergence Validation Time

Definition:

The amount of time for which the completion of Full Convergence is maintained without additional Impaired Packets being observed.

Discussion:

The purpose of the Sustained Convergence Validation Time is to produce convergence benchmarks protected against fluctuation in Forwarding Rate after the completion of Full Convergence is observed. The RECOMMENDED Sustained Convergence Validation Time to be used is the time to send 5 consecutive packets to each destination with a minimum of 5 seconds. The Benchmarking Methodology Working Group (BMWG) selected 5 seconds based upon [Br99] which recommends waiting 2 seconds for residual frames to arrive (this is the Forwarding Delay Threshold for the last packet sent) and 5 seconds for DUT restabilization.

Measurement Units: seconds (and fractions)

See Also:

Full Convergence, Convergence Recovery Instant

3.7.6. Forwarding Delay Threshold

Definition:

The maximum waiting time threshold used to distinguish between packets with very long delay and lost packets that will never arrive.

Discussion:

Applying a Forwarding Delay Threshold allows to consider packets with a too large Forwarding Delay as being lost, as is required for some applications (e.g. voice, video, etc.). The Forwarding Delay Threshold is a parameter of the methodology, and it MUST be reported. [Br99] recommends waiting 2 seconds for residual frames to arrive.

Measurement Units: seconds (and fractions)

See Also:

Convergence Packet Loss, Connectivity Packet Loss

3.8. Miscellaneous Terms

3.8.1. Impaired Packet

Definition:

A packet that experienced at least one of the following impairments: loss, excessive Forwarding Delay, corruption, duplication, reordering.

Discussion:

A lost packet, a packet with a Forwarding Delay exceeding the Forwarding Delay Threshold, a corrupted packet, a Duplicate Packet [Po06], and an Out-of-Order Packet [Po06] are Impaired Packets.

Packet ordering is observed for each individual flow (See [Th00], Section 3) of the Offered Load.

Measurement Units: N/A

See Also: Forwarding Delay Threshold

4. Security Considerations

Benchmarking activities as described in this memo are limited to

technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the constraints specified in the sections above.

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT/SUT.

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

5. IANA Considerations

This document requires no IANA considerations.

6. Acknowledgements

Thanks to Sue Hares, Al Morton, Kevin Dubray, Ron Bonica, David Ward, Peter De Vriendt, Anuj Dewagan, Adrian Farrel, Stewart Bryant, Francis Dupont, and the Benchmarking Methodology Working Group for their contributions to this work.

7. Normative References

- [Br91] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [Br97] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [Br99] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [Ca90] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, December 1990.
- [Co08] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, July 2008.

- [De02] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, November 2002.
- [Ho08] Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, October 2008.
- [Ko02] Koodli, R. and R. Ravikanth, "One-way Loss Pattern Sample Metrics", RFC 3357, August 2002.
- [Ma98] Mandeville, R., "Benchmarking Terminology for LAN Switching Devices", RFC 2285, February 1998.
- [Mo98] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.
- [Po06] Poretsky, S., Perser, J., Erramilli, S., and S. Khurana, "Terminology for Benchmarking Network-layer Traffic Control Mechanisms", RFC 4689, October 2006.
- [Pollm] Poretsky, S., Imhoff, B., and K. Michielsen, "Benchmarking Methodology for Link-State IGP Data Plane Route Convergence", draft-ietf-bmwg-igp-dataplane-conv-meth-23 (work in progress), January 2011.
- [Th00] Thaler, D. and C. Hopps, "Multipath Issues in Unicast and Multicast Next-Hop Selection", RFC 2991, November 2000.

Authors' Addresses

Scott Poretsky
Allot Communications
67 South Bedford Street, Suite 400
Burlington, MA 01803
USA

Phone: + 1 508 309 2179
Email: sporetsky@allot.com

Brent Imhoff
Juniper Networks
1194 North Mathilda Ave
Sunnyvale, CA 94089
USA

Phone: + 1 314 378 2571
Email: bimhoff@planetispork.com

Kris Michielsen
Cisco Systems
6A De Kleetlaan
Diegem, BRABANT 1831
Belgium

Email: kmichiel@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 22, 2012

A. Morton
AT&T Labs
October 20, 2011

IMIX Genome: Specification of variable packet sizes for additional
testing
draft-ietf-bmwg-imix-genome-00

Abstract

Benchmarking Methodologies have always relied on test conditions with constant packet sizes, with the goal of understanding what network device capability has been tested. Tests with constant packet size reveal device capabilities but differ significantly from the conditions encountered in operational deployment, and so additional tests are sometimes conducted with a mixture of packet sizes, or "IMIX". The mixture of sizes a networking device will encounter is highly variable and depends on many factors. An IMIX suited for one networking device and deployment will not be appropriate for another. However, the mix of sizes may be known and the tester may be asked to augment the fixed size tests. To address this need, and the perpetual goal of specifying repeatable test conditions, this draft defines a way to specify the exact repeating sequence of packet sizes from the usual set of fixed sizes, and other forms of mixed size specification.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

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 <http://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 April 22, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	4
2. Scope and Goals	4
3. Specification of the IMIX Genome	5
4. Specification of a Custom IMIX	6
5. Reporting Long or Pseudo-Random Packet Sequences	7
6. Security Considerations	7
7. IANA Considerations	8
8. Acknowledgements	8
9. References	8
9.1. Normative References	8
9.2. Informative References	8
Author's Address	9

1. Introduction

This memo defines a method to unambiguously specify the sequence of packet sizes used in a load test.

Benchmarking Methodologies [RFC2544] have always relied on test conditions with constant packet sizes, with the goal of understanding what network device capability has been tested. Tests with the smallest size stress the header processing capacity, and tests with the largest size stress the overall bit processing capacity. Tests with sizes in-between may determine the transition between these two capacities.

Streams of constant packet size differ significantly from the conditions encountered in operational deployment, and so additional tests are sometimes conducted with a mixture of packet sizes. The set of sizes used is often called an Internet Mix, or "IMIX" [Spirent], [IXIA], [Agilent].

The mixture of sizes a networking device will encounter is highly variable and depends on many factors. An IMIX suited for one networking device and deployment will not be appropriate for another. However, the mix of sizes may be known and the tester may be asked to augment the fixed size tests. The references above cite the original studies and their methodologies - similar methods can be used to determine new size mixes.

To address this need, and the perpetual goal of specifying repeatable test conditions, this draft proposes a way to specify the exact repeating sequence of packet sizes from the usual set of fixed sizes: the IMIX Genome. Other, less exact forms of size specification are also recommended for extremely complicated or customized size mixes.

This memo takes the position that it cannot be proven for all circumstances that the sequence of packet sizes does not affect the test result, thus a standardized specification of sequence is valuable.

2. Scope and Goals

This memo defines a method to unambiguously specify the sequence of packet sizes that have been used in a load test, assuming that a relevant mix of sizes is known to the tester and the length of the repeating sequence is not very long (<100 packets).

The IMIX Genome will allow an exact sequence of packet sizes to be communicated as a single-line name, resolving the current ambiguity

with results that simply refer to "IMIX".

While documentation of the exact sequence is ideal, the memo also covers the case where the sequence of sizes is very long or may be generated by a pseudo-random process.

It is a colossal non-goal to standardize one or more versions of the IMIX. This topic has been discussed on many occasions on the `bmwg-list[IMIXonList]`. The goal is to enable customization with minimal constraints while fostering repeatable testing once the fixed size testing is complete.

3. Specification of the IMIX Genome

The IMIX Genome is specified in the following format:

IMIX - 123456...x

where each number is replaced by the letter corresponding to the size of the packet at that position in the sequence. The following table gives the letter encoding for the [RFC2544] standard sizes (64, 128, 256, 512, 1024, 1280, and 1518 bytes) and "jumbo" sizes (2112, 9000, 16000). Note that the 4 octet Ethernet frame check sequence may fail to detect bit errors in the larger jumbo frames, see [jumbo].

Size, bytes	Genome Code Letter
64	a
128	b
256	c
512	d
1024	e
1280	f
1518	g
2112	h
9000	i
16000	j
MTU	z

For example: a five packet sequence with sizes 64,64,64,1280,1518 would be designated:

IMIX - aaafg

While this approach allows some flexibility, there are also

constraints.

- o Non-RFC2544 packet sizes would need to be approximated by those available in the table.
- o The Genome for very long sequences can become undecipherable by humans.
- o z=MTU is seen as valuable (so far).
- o Whether more tabulated packet sizes would be useful is TBD, and "jumbo" sizes were added in this version.

Some questions testers must ask and answer when using the IMIX Genome are:

1. Multiple Source-Destination Address Pairs: is the IMIX sequence applicable to each pair, across multiple pairs in sets, or across all pairs?
2. Multiple Tester Ports: is the IMIX sequence applicable to each port, across multiple ports in sets, or across all ports?

The chosen configuration would be expressed the following general form:

Source Address/Port/Blade	Destination Address/Port/Blade	Corresponding IMIX
x.x.x.x Blade2	y.y.y.y Blade3	IMIX - aaafg

where testers can specify the IMIX used between any two entities in the test architecture.

4. Specification of a Custom IMIX

The Custom IMIX is specified in the following format:

CUSTOM IMIX - 123456...x

where each number is replaced by the letter corresponding to the size of the packet at that position in the sequence. The tester MUST complete the following table, giving the letter encoding for each size used, where each set of three lower-case letters would be replaced by the integer size in octets.

Size, bytes	Custom Code Letter
aaa	A
bbb	B
ccc	C
ddd	D
eee	E
fff	F
ggg	G
etc.	up to Z

For example: a five packet sequence with sizes aaa,aaa,aaa,ggg,ggg would be designated:

CUSTOM IMIX - AAAGG

5. Reporting Long or Pseudo-Random Packet Sequences

When the IMIX-Genome cannot be used (when the sheer length of the sequence would make the genome unmanageable) or when the sequence is designed to vary within some proportional constraints, a table is necessary.

IP Length	Percentage of Total	Other Length(s)
64	23	82
128	67	146
1000	10	1018

Note that this approach also allows non-standard packet sizes, but trades the short genome specification and ability to specify the exact sequence for other flexibilities.

6. Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the other constraints [RFC2544].

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test

traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT/SUT.

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

7. IANA Considerations

This memo makes no requests of IANA, and hopes that IANA will leave it alone as well.

8. Acknowledgements

Thanks to Sarah Banks, Aamer Akhter, and Steve Maxwell for their review and comments.

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.

9.2. Informative References

- [Agilent] http://www.ixiacom.com/pdfs/test_plans/agilent_journal_of_internet_test_methodologies.pdf, "The Journal of Internet Test Methodologies", 2007.
- [IMIXonList] <http://www.ietf.org/mail-archive/web/bmwg/current/msg00691.html>, "Discussion on IMIX", 2003.
- [IXIA] http://www.ixiacom.com/library/test_plans/display?skey=testing_pppox, "Library: Test Plans", 2010.
- [Spirent] <http://gospirent.com/whitepaper/>

IMIX%20Test%20Methodolgy%20Journal.pdf, "Test Methodology
Journal: IMIX (Internet Mix) Journal", 2006.

[jumbo] [http://www.ietf.org/mail-archive/web/bmwg/current/
msg00691.html](http://www.ietf.org/mail-archive/web/bmwg/current/msg00691.html), "Discussion of Jumbo Packets and FCS
Failure, see <http://sd.wareonearth.com/~phil/jumbo.html>
and <http://staff.psc.edu/mathis/MTU/arguments.html#crc>".

Author's Address

Al Morton
AT&T Labs
200 Laurel Avenue South
Middletown,, NJ 07748
USA

Phone: +1 732 420 1571
Fax: +1 732 368 1192
Email: acmorton@att.com
URI: <http://home.comcast.net/~acmacm/>

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: 2 April, 2012

Jan Novak
Cisco Systems, Inc.
2 October 2011

IP Flow Information Accounting and Export Benchmarking
Methodology
draft-ietf-bmwg-ipflow-meth-04.txt

Abstract

This document provides a methodology and framework for quantifying the performance impact of monitoring of IP flows on a network device and export of this information to a collector. It identifies the rate at which the IP flows are created, expired, and successfully exported as a new performance metric in combination with traditional throughput. The metric is only applicable to the devices compliant with the Architecture for IP Flow Information Export [RFC5470].

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and 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 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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on 2 April, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

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 RFC 2119 [RFC2119].

Table of Contents

1. Introduction.	3
2. Terminology	4
2.1 Existing Terminology.	4
2.2 New Terminology	4
3. Flow Monitoring Performance Benchmark	6
3.1 Definition.	6
3.2 Device Applicability.	6
3.3 Measurement Concept	7
3.4 The Measurement Procedure Overview.	8
4. Measurement Set Up.	9
4.1 Measurement Topology.	9
4.2 Base DUT Set Up.	11
4.3 Flow Monitoring Configuration.	11
4.4 Collector.	15
4.5 Sampling	16
4.6 Frame Formats.	16
4.7 Frame Sizes.	16
4.8 Flow Export Data Packet Sizes.	16
4.9 Illustrative Test Set-up Examples.	17
5. Flow Monitoring Throughput Measurement Methodology	18
5.1 Flow Monitoring Configuration.	18
5.2 Traffic Configuration.	19
5.3 Cache Population	19
5.4 Measurement Time Interval.	20
5.5 Flow Export Rate Measurement	21
5.6 The Measurement Procedure.	21
6. RFC2544 Measurements	22
6.1 Flow Monitoring Configuration.	23
6.2 Measurements With the Flow Monitoring Throughput Set-up.	23
6.3 Measurements With Fixed Flow Export Rate	23
6.4 Measurements With Single Traffic Component	23
6.5 Measurements With Two Traffic Components	24
7. Flow Monitoring Accuracy	25
8. Evaluating Flow Monitoring Applicability	25
9. Acknowledgements	26
10. Security Considerations	26
11. IANA Consierations.	26
12. References.	26
12.1 Normative References.	26
12.2 Informative References.	27
Appendix A: Recommended Report Format	28

Appendix B: Miscellaneous Tests	29
B.1 DUT Under Traffic Load	29
B.2 In-band Flow Export.	29
B.3 Variable Packet Rate	30
B.4 Bursty Traffic	30
B.5 Various Flow Monitoring Configurations	30
B.6 Tests With Bidirectional Traffic	31
B.7 Instantaneous Flow Export Rate	31

1. Introduction

Monitoring of IP flows (Flow monitoring) is defined in the Architecture for IP Flow Information Export [RFC5470] and related IPFIX documents. It analyses the traffic using predefined fields from the packet header as keys and stores the traffic and other internal information in the DUT (Device Under Test) memory. This cached flow information is then formatted into records (see section 2.1 for term definitions) and exported from the DUT to an external data collector for analysis. More details on the measurement architecture is provided in section 3.3.

Flow monitoring on network devices is widely deployed and has numerous uses in both service provider and enterprise segments as detailed in the Requirements for IP Flow Information Export [RFC3917]. This document provides a methodology for measuring Flow monitoring performance so that network operators have a framework for considering measurement impact on the network and network equipment.

This document's goal is a series of methodology specifications for the measurement of Flow monitoring performance, in a way that is comparable amongst various implementations, platforms, and vendor's devices.

Since Flow monitoring will in most cases run on network devices also forwarding packets, the methodology for [RFC2544] measurements (with IPv6 and MPLS specifics defined in [RFC5180] and [RFC5695] respectively) in the presence of Flow monitoring is also employed here.

The most significant performance parameter is the rate at which IP flows are created and expired in the network device's memory and exported to a collector. Therefore, this document specifies a methodology to measure the maximum IP flow rate that a network device can sustain without impacting the forwarding plane, without losing any IP flow information, and without compromising the IP flow accuracy (see section 7 for details).

[RFC2544], [RFC5180] and [RFC5695] specify benchmarking of network devices forwarding IPv4, IPv6 and MPLS [RFC3031] traffic, respectively. The methodology specified in this document stays the same for any traffic type. The only restriction may be the DUT's lack of support for Flow monitoring of the particular traffic type.

A variety of different network device architectures exist that are capable of Flow monitoring and export. As such, this document does not attempt to list the various white box variables (CPU load, memory utilization, hardware resources utilization etc) that could be gathered as they always help in comparison evaluations. A more complete understanding of the stress points of a particular device can be attained using this internal information and the tester MAY choose to gather this information during the measurement iterations.

2. Terminology

The terminology used in this document is based on [RFC5470], [RFC2285] and [RFC1242] as summarised in section 2.1. The only new terms needed for this methodology are defined in section 2.2.

2.1 Existing Terminology

Device Under Test (DUT)	[RFC2285, section 3.1.1]
Flow	[RFC5470, section 2]
Flow Key	[RFC5470, section 2]
Flow Record	[RFC5470, section 2]
Observation Point	[RFC5470, section 2]
Metering Process	[RFC5470, section 2]
Exporting Process	[RFC5470, section 2]
Exporter	[RFC5470, section 2]
Collector	[RFC5470, section 2]
Control Information	[RFC5470, section 2]
Data Stream	[RFC5470, section 2]
Flow Expiration	[RFC5470, section 5.1.1]
Flow Export	[RFC5470, section 5.1.2]
Throughput	[RFC1242, section 3.17]

2.2 New Terminology

2.2.1 Cache

Definition:

Memory area held and dedicated by the DUT to store Flow information prior to the Flow Expiration.

2.2.2 Cache Size

Definition:

The size of the Cache in terms of how many entries the Cache can hold.

Discussion:

This term is typically represented as a configurable option in the particular Flow monitoring implementation. Its highest value will depend on the memory available in the network device.

Measurement units:

Number of Cache entries

2.2.3 Active Timeout

Definition:

For long-running Flows, the time interval after which the Metering Process expires a Cache entry to ensure Flow data is regularly updated

Discussion:

This term is typically presented as a configurable option in the particular Flow monitoring implementation. See section 5.1.1 of [RFC5470] for more detailed discussion.

Flows are considered long-running when they last longer than several multiples of the Active Timeout or when the Active Timeout is zero, contain a larger number of packets than usual for a single transaction based Flows, in the order of tens of packets and higher.

Measurement units:

Seconds

2.2.4 Inactive Timeout

Definition:

The time interval used by the Metering Process to expire an entry from the Cache, when no more packets belonging to that specific Cache entry have been observed during the interval.

Discussion:

This term is typically represented as a configurable option in the particular Flow monitoring implementation. See section 5.1.1 of [RFC5470] for more detailed discussion.

Measurement units:

Seconds

2.2.5 Flow Export Rate

Definition:

The number of Cache entries that expire from the Cache (as defined by the Flow Expiration term) and are exported to the Collector within a measurement time interval. There SHOULD NOT be any export filtering, so that all the expired cache entries are exported. If there is export filtering and it can't be disabled, this needs to be noted.

The measured Flow Export Rate MUST include *both* the Data Stream and the Control Information, as defined in section 2 of [RFC5470].

Discussion:

The Flow Export Rate is measured using Flow Export data observed at the Collector by counting the exported Flow Records during the measurement time interval (see section 5.4). The value obtained is an average of the instantaneous export rates observed during the measurement time interval. The smallest possible measurement interval (if attempting to measure nearly instantaneous export rate rather than average export rate on the DUT) is limited by the export capabilities of the particular Flow monitoring implementation (when possible physical layer issues between the DUT and the Collector are excluded).

Measurement units:

Number of Flow Records per second

3. Flow Monitoring Performance Benchmark

3.1 Definition

Flow Monitoring Throughput

Definition:

The maximum Flow Export Rate the DUT can sustain without losing a single Cache entry. Additionally, for packet forwarding devices, the maximum Flow Export Rate the DUT can sustain without dropping packets in the Forwarding Plane (see figure 1).

Measurement units:

Number of Flow Records per second

Discussion:

The losses of Cache entries or forwarded packets in this definition are assumed to happen due to the lack of DUT resources to process any additional traffic information or lack of resources to process Flow Export data. The physical layer issues, like insufficient bandwidth from the DUT to the Collector or lack of Collector resources MUST be excluded as detailed in section 4.

3.2 Device Applicability

The Flow monitoring performance metric is applicable to network devices that implement [RFC5470] architecture. These devices can be network packet forwarding devices or appliances which analyze

the traffic but do not forward traffic (probes, sniffers, replicators).

This document does not intend to measure Collector performance, it only requires sufficient Collector resources (as specified in section 4.4) in order to measure the DUT characteristics.

3.3 Measurement Concept

Figure 1 below presents the functional block diagram of the DUT. The traffic in the figure represents the test traffic sent to the DUT and forwarded by the DUT, if possible. When testing devices which do not act as network packet forwarding devices (such as probes, sniffers and replicators) the forwarding plane is simply an Observation Point as defined in section 2 of [RFC5470]. The [RFC2544] Throughput of such devices will always be zero and the only applicable performance metric is the Flow Monitoring Throughput.

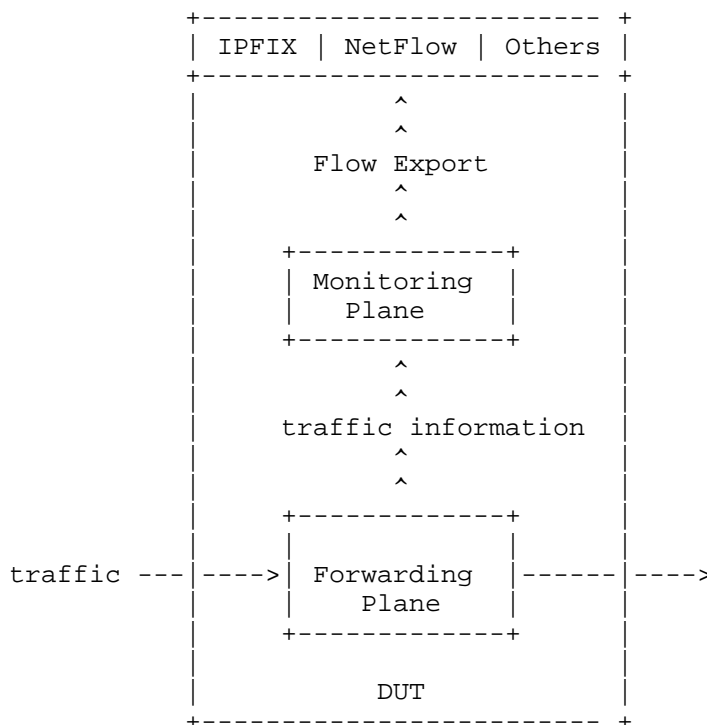


Figure 1. The functional block diagram of the DUT

The Flow monitoring enabled (see section 4.3) on the DUT and represented in the figure 1 by the Monitoring Plane uses the traffic information provided by the Forwarding Plane and configured Flow Keys to create Cache entries representing the traffic forwarded (or observed) by the DUT in the DUT Cache. The Cache entries are expired from the Cache depending on the Cache configuration (ie, the Active and Inactive Timeouts, number of Cache

entries and the Cache Size) and the traffic pattern. The Cache entries are used by the Exporting Process to format the Flow Records which are then exported from the DUT to the Collector (see figure 2 in section 4).

The Forwarding Plane and Monitoring Plane represent two separate functional blocks, each with its own performance capability. The Forwarding Plane handles user data packets and is fully characterised by the metrics defined by [RFC2544].

The Monitoring Plane handles Flows which reflect the analysed traffic. The metric for Monitoring Plane performance is Flow Export Rate, and the benchmark is the Flow Monitoring Throughput.

3.4 The Measurement Procedure Overview

The measurement procedure is fully specified in sections 4, 5 and 6. This section provides an overview of principles for the measurements.

The basic measurement procedure of performance characteristics of a DUT with Flow monitoring enabled is a conventional Throughput measurement using a search algorithm to determine the maximum packet rate at which none of the offered packets and corresponding Flow Records are dropped by the DUT as described in [RFC1242] and section 26.1 of [RFC2544].

The Device Under Test (DUT) with Flow monitoring enabled contains two functional blocks which need to be measured using characteristics applicable to one or both blocks (see figure 1). See sections 3.4.1 and 3.4.2 for further discussion.

On one hand the Monitoring Plane and Forwarding Plane (see figure 1) need to be looked at as two independent blocks, and the performance of each of them measured independently. But on the other hand when measuring the performance of one of them, the status and performance of the other MUST be known and benchmarked when both are present.

3.4.1 Monitoring Plane Performance Measurement

The Flow Monitoring Throughput MUST be (and can only be) measured with one packet per Flow as specified in section 5. This traffic type represents the most demanding traffic from the Flow monitoring point of view and will exercise the Monitoring Plane (see figure 1) of the DUT most. In this scenario every packet seen by DUT creates a new Cache entry and forces the DUT to fill the Cache instead of just updating packet and byte counters of an already existing Cache entry.

The exit criteria for the Flow Monitoring Throughput measurement are one of the following (e.g. if any of the conditions is reached):

- a. The Flow Export Rate at which the DUT starts to lose Flow information or the Flow information gets corrupted
- b. The Flow Export Rate at which the Forwarding Plane starts to drop or corrupt packets (if the Forwarding Plane is present)

A corrupted packet here means the packet header corruption (resulting in the cyclic redundancy check failure on the transmission level and consequent packet drop) or the packet payload corruption leading to the lost application level data.

3.4.2 Forwarding Plane Performance Measurement

The Forwarding Plane (see figure 1) performance metrics are fully specified by [RFC2544] and MUST be measured accordingly. A detailed traffic analysis (see below) with relation to Flow monitoring MUST be performed prior of any [RFC2544] measurements. Mainly the Flow Export Rate caused by the test traffic during an [RFC2544] measurement MUST be known and reported.

The required test traffic analysis mainly involves the following:

- a. Which packet header parameters are incremented or changed during traffic generation
- b. Which Flow Keys the Flow monitoring configuration uses to generate Flow Records

The RFC2544 performance metrics can be measured in one of the three modes:

- a. As a baseline of forwarding performance without Flow monitoring
- b. At a certain level of Flow monitoring activity specified by a Flow Export Rate lower than the Flow Monitoring Throughput
- c. At the maximum level of Flow monitoring performance, e.g. using traffic conditions representing a measurement of Flow Monitoring Throughput

The above mentioned measurement mode in point a. represents an ordinary Throughput measurement specified in RFC2544. The details how to setup the measurements in points b. and c. are given in section 6.

4. Measurement Set Up

This section concentrates on the set-up of all components necessary to perform Flow monitoring performance measurement. The recommended reporting format can be found in Appendix A.

4.1 Measurement Topology

The measurement topology described in this section is applicable only to the measurements with packet forwarding network devices. The possible architectures and implementation of the traffic monitoring appliances (see section 3.2) are too various to be covered in this

document. Instead of the Forwarding Plane, these appliances generally have some kind of feed (an optical splitter, an interface sniffing traffic on a shared media or an internal channel on the DUT providing a copy of the traffic) providing the information about the traffic necessary for Flow monitoring analysis. The measurement topology then needs to be adjusted to the appliance architecture, and MUST be part of the measurement report.

The measurement set-up is identical to that used by [RFC2544], with the addition of a Collector to analyze the Flow Export(see figure 2).

In the measurement topology with unidirectional traffic, the traffic is transmitted from the sender to the receiver through the DUT. The received traffic is analyzed to check it is identical to the generated traffic.

The ideal way to implement the measurement is by using a single device to provide the sender and receiver capabilities with a sending port and a receiving port. This allows for an easy check whether all the traffic sent by the sender was re-transmitted by the DUT and received at the receiver.

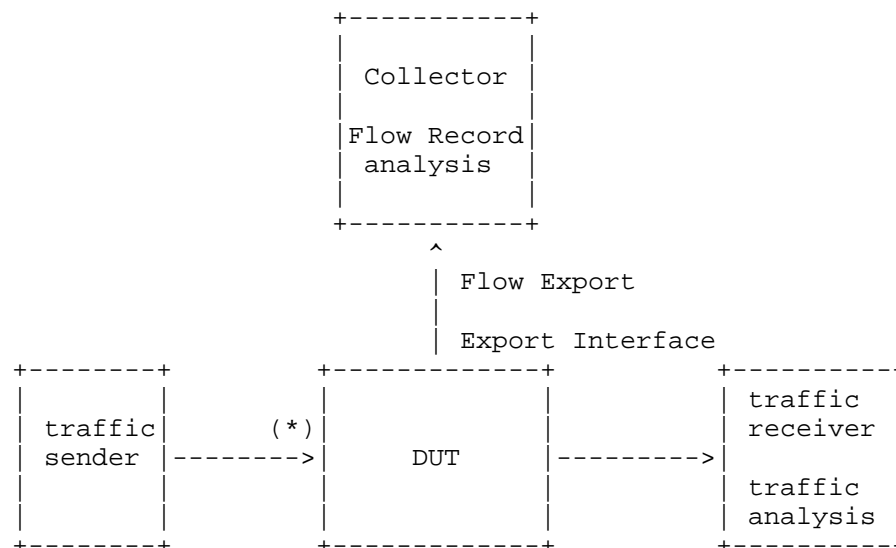


Figure 2 Measurement topology with unidirectional traffic

The DUT's export interface (connecting the Collector) MUST NOT be used for forwarding the test traffic but only for the Flow Export data containing the Flow Records. In all measurements, the export interface MUST have enough bandwidth to transmit Flow Export data without congestion. In other words, the export interface MUST NOT be a bottleneck during the measurement.

Note that more complex topologies might be required. For example, if the effects of enabling Flow monitoring on several interfaces are of concern or the media maximum speed is less than the DUT throughput,

the topology can be expanded with several input and output ports. However, the topology MUST be clearly written in the measurement report.

4.2 Baseline DUT Set Up

The baseline DUT set-up and the way the set-up is reported in the measurement results is fully specified in section 7 of [RFC2544].

The baseline DUT configuration might include other features like packet filters or quality of service on the input and/or output interfaces if there is the need to study Flow monitoring in the presence of those features. The Flow monitoring measurement procedures do not change in this case. Consideration needs to be made when evaluating measurement results to take into account the possible change of packet rates offered to the DUT and Flow monitoring after application of the features to the configuration. Any such feature configuration MUST be part of the measurement report.

The DUT export interface (see figure 2) MUST be configured with sufficient output buffers to avoid dropping the Flow Export data due to a simple lack of resources in the interface hardware. The applied configuration MUST be part of the measurement report.

4.3 Flow Monitoring Configuration

This section covers all the aspects of the Flow monitoring configuration necessary on the DUT in order to perform the Flow monitoring performance measurement. The necessary configuration has a number of components (see [RFC5470]), namely Observation Points, Metering Process and Exporting Process as detailed below.

The DUT MUST support the Flow monitoring architecture as specified by [RFC5470]. The DUT SHOULD support IPFIX [RFC5101].

The DUT configuration and any existing Cache MUST be erased before application of any new configuration for the currently executed measurement.

4.3.1 Observation Points

The Observation Points specify the interfaces and direction where the Flow monitoring traffic analysis is to be performed.

The (*) in Figure 2 designates the Observation Points in the default configuration. Other DUT Observation Points might be configured depending on the specific measurement needs as follows:

- a. ingress port/ports(s) only
- b. egress port(s) /ports only
- c. both ingress and egress

Generally, the placement of Observation Points depends upon the position of the DUT in the deployed network and the purpose of Flow monitoring. See [RFC3917] for detailed discussion. The measurement procedures are otherwise the same for all these possible configurations.

In the case when both ingress and egress Flow monitoring is enabled on one DUT the results analysis needs to take into account that each Flow will be represented in the DUT Cache by two Flow Records (one for each direction) and therefore also the Flow Export will contain those two Flow Records.

If more than one Observation Point for one direction is defined on the DUT the traffic passing through each of the Observation Points MUST be configured in such a way that it creates Flows and Flow Records which do not overlap, e.g. each packet (or set of packets if measuring with more than one packet per Flow - see section 6.4) sent to the DUT on different ports still creates one unique Flow Record.

The specific Observation Points and associated monitoring direction MUST be included as part of the report of the results.

4.3.2 Metering Process

The Metering Process MUST be enabled in order to create the Cache in the DUT and configure the Cache related parameters.

The Cache Size available to the DUT MUST be known and taken into account when designing the measurement as specified in section 5.

The Cache's Inactive and Active Timeouts MUST be known and taken into account when designing the measurement as specified in section 5. If the Flow monitoring implementation allows only timeouts zero (e.g. immediate timeout or non-existent Cache) then the measurement conditions in section 5 are fulfilled inherently without any additional configuration. The DUT simply instantly exports information about every single packet.

If the Flow monitoring implementation allows to configure multiple Metering Processes on a single DUT, the exact configuration of each process MUST be included in the results report. Only measurements with the same number of Metering Processes can be compared.

The Cache Size, the Inactive and Active Timeouts MUST be included as part of the results report.

4.3.3 Exporting Process

The Exporting Process MUST be configured in order to export the Flow Record data to the Collector.

The Exporting Process MUST be configured in such a way that all Flow Records from all configured Observation Points are exported towards the Collector, after the expiration policy composed of the Inactive and Active Timeouts and Cache Size.

Novak

Expires April, 2012

[Page 12]

Internet-Draft

Flow Monitoring Benchmarking

October 2011

The Exporting Process SHOULD be configured with IPFIX [RFC5101] as the protocol to use to format the Flow Export data. If the Flow monitoring implementation does not support it, proprietary protocols MAY be used. Only measurements with same export protocol SHOULD be compared since the protocols may differ in their export efficiency.

Various Flow monitoring implementations might use different default values regarding the export of Control Information [RFC5470] and therefore Flow Export corresponding to Control Information SHOULD be analyzed and reported as a separate item on the measurement report. Preferably, the export of Control Information SHOULD always be configured consistently across all testing and configured to the minimal possible value - ideally just one exported set of Control Information during each measurement. Note that Control Information includes IPFIX Options and Templates [RFC5101].

Section 10 of [RFC5101] and section 8.1 of [RFC5470] discuss the possibility of deploying various transport layer protocols to deliver Flow Export data from the DUT to the Collector. The selected protocol MUST be included in the measurement report. Only benchmarks with the same transport layer protocol should be compared. If the Flow monitoring implementation allows the use of multiple the transport layer protocols, each of the protocols SHOULD be measured in a separate measurement run and the results reported independently in the report.

If a reliable transport protocol is used for the transmission of the Flow Export data from the DUT, the configuration of the Transport session MUST allow for non-blocking data transmission. An example of parameters to look at would be TCP window size and maximum segment size (MSS). The most substantial transport layer parameters should be included in the report.

4.3.4 Flow Records

A Flow Record contains information about a specific Flow that was observed at an Observation Point. A Flow Record contains measured properties of the Flow (e.g., the total number of bytes for all the Flow's packets) and usually characteristic properties of the Flow (e.g., source IP address).

The Flow Record definition is implementation specific. A Flow monitoring implementation might allow for only a fixed Flow Record definition, based on the most common IP parameters in the IPv4 or IPv6 headers - for example source and destination IP addresses, IP protocol numbers or transport level port numbers. Another implementation might allow the user to define their own arbitrary Flow Record to monitor the traffic. The requirement for the measurements defined in this document is only the need for a large number of Cache entries in the Cache. The Flow Keys needed to achieve that will typically be source and destination IP addresses and transport level port numbers.

The recommended full IPv4, IPv6 or MPLS Flow Record is shown below:

Flow Keys:

- Source IP address
- Destination IP address
- MPLS label (for MPLS traffic type only)
- Transport layer source port

- Transport layer destination port
- IP protocol number (IPv6 next header)
- IP type of service (IPv6 traffic class)

Other fields:

- Packet counter
- Byte counter

Table 1: Recommended Configuration

If the Flow monitoring allows for user defined Flow Records, the minimal Flow Record configurations allowing large numbers of Cache entries for example are:

Flow Keys:

- Source IP address
- Destination IP address

Other fields:

- Packet counter

or:

Flow Key fields

- Transport layer source port
- Transport layer destination port

Other fields

- Packet counter

Table 2: User-defined Configuration

The Flow Record configuration MUST be clearly noted in the measurement report. The Flow Monitoring Throughput measurements on different DUTs or different Flow monitoring implementations MUST be compared only for exactly the same Flow Record configuration.

4.3.5 Flow Monitoring With Multiple Configurations

The Flow monitoring architecture as specified in [RFC5470] allows for more complicated configurations with multiple Metering and Exporting Processes on a single DUT. Depending on the particular Flow monitoring implementation it might affect the measured DUT performance. The test report should therefore contain information containing how many Metering and Exporting processes were configured on the DUT for the selected Observation Points.

The examples of such possible configurations are:

- a. Several Observation Points with a single Metering Process and a single Exporting Process
- b. Several Observation Points, each with one Metering Process but all using just one instance of Exporting Process
- c. Several Observation Points with per Observation Point Metering Process and Exporting Process

4.3.6 MPLS Measurement Specifics

The Flow Record configuration for measurements with MPLS encapsulated traffic SHOULD contain the MPLS label.

The tester SHOULD ensure that the data received by the Collector contains the expected MPLS labels.

The MPLS forwarding performance document [RFC5695] specifies a number of possible MPLS label operations to test. The Observation Points MUST be placed on all the DUT test interfaces where the particular MPLS label operation takes place. The performance measurements SHOULD be performed with only one MPLS label operation at the time.

The DUT MUST be configured in such a way that all the traffic is subject to the measured MPLS label operation.

4.4 Collector

The Collector is needed in order to capture the Flow Export data which allows the Flow Monitoring Throughput to be measured.

The Collector can be used as exclusively capture device providing just hexadecimal format of the Flow Export data. In such a case it does not need to have any additional Flow Export decoding capabilities and all the decoding is done off line.

However if the Collector is also used to decode the Flow Export data then it SHOULD support IPFIX [RFC5101] for easier results analysis. If proprietary Flow Export is deployed, the Collector MUST support it otherwise the Flow Export data analysis is not possible.

The Collector MUST be capable of capturing at the full rate the export packets sent from the DUT without losing any of them. In the case of the use of reliable transport protocols (see also section 4.3.3) to transmit Flow Export data, the Collector MUST have sufficient resources to guarantee non-blocking data transmission on the transport layer session.

During the analysis, the Flow Export data needs to be decoded and the received Flow Records counted.

The capture buffer MUST be cleared at the beginning of each measurement.

4.5 Sampling

Packet sampling and flow sampling is out of scope of this document. This document applies to situations without packet or flow sampling.

4.6 Frame Formats

Flow monitoring itself is not dependent in any way on the media used on the input and output ports. Any media can be used as supported by the DUT and the test equipment.

At the time of writing the most common transmission media and corresponding frame formats (Ethernet, Packet over SONET) for IPv4, IPv6 and MPLS traffic are specified within [RFC2544], [RFC5180] and [RFC5695].

The presented frame formats MUST be recorded in the report.

4.7 Frame Sizes

Frame sizes of the traffic to be analyzed by the DUT are specified in [RFC2544] section 9 for Ethernet type interfaces (64, 128, 256, 1024, 1280, 1518 bytes) and in [RFC5180] section 5 for Packet over SONET interfaces (47, 64, 128, 256, 1024, 1280, 1518, 2048, 4096 bytes).

When measuring with large frame sizes, care needs to be taken to avoid any packet fragmentation on the DUT interfaces which could negatively affect measured performance values.

The presented frame sizes MUST be recorded in the report.

4.8 Flow Export Data Packet Sizes

The Flow monitoring performance will be affected by the packet size the particular implementation uses to transmit Flow Export data to the Collector. The used packet size SHOULD be part of the test report and only measurements with same packet sizes SHOULD be compared.

The DUT export interface (see figure 2) maximum transmission unit (MTU) SHOULD be configured to the largest available value for the media. The MTU MUST be recorded in the report.

4.9 Illustrative Test Set-up Examples

The below examples represent a hypothetical test set-up to clarify the use of Flow monitoring parameters and configuration, together with traffic parameters to test Flow monitoring. The actual benchmarking specifications are in sections 5 and 6.

4.9.1 Example 1 - Inactive Timeout Flow Expiration

The traffic generator sends 1000 packets per second in 10000 defined streams, each stream identified by an unique destination IP address. Therefore each stream has a packet rate of 0.1 packets per second.

The packets are sent in a round robin fashion (stream 1 to 10000) while incrementing the destination IP address for each sent packet.

The configured Cache Size is 20000 Flow Records. The configured Active Timeout is 100 seconds, the Inactive Timeout is 5 seconds.

Flow monitoring on the DUT uses the destination IP address as the Flow Key.

A packet with destination IP address equal to A is sent every 10 seconds, so the Cache entry would be refreshed in the Cache every 10 seconds. However, the Inactive Timeout is 5 seconds, so the Cache entries will expire from the Cache due to the Inactive Timeout and when a new packet is sent with the same IP address A it will create a new entry in the Cache. This behaviour depends upon the design an efficiency of the cache ager, and incidences of multi-packet flows observed during this test should be noted.

The measured Flow Export Rate in this case will be 1000 Flow Records per second since every single sent packet will always create a new Cache entry and we send 1000 packets per second.

The expected number of Cache entries in the Cache during the whole measurement is around 5000. It corresponds to the Inactive Timeout being 5 seconds and during those five seconds 5000 entries are created. This expectation might change in real measurement set-ups with large Cache Sizes and high packet rate where the DUT's actual export rate might be limited and lower than the Flow Expiration activity caused by the traffic offered to the DUT. This behaviour is entirely implementation specific.

4.9.2 Example 2 - Active Timeout Flow Expiration

The traffic generator sends 1000 packets per second in 100 defined streams, each stream identified by an unique destination IP address. So each stream has a packet rate of 10 packets per second. The packets are sent in a round robin fashion (stream 1 to 100) while incrementing the destination IP address for each sent packet.

The configured Cache Size is 1000 Flow Records. The configured Active Timeout is 100 seconds. The Inactive Timeout is 10 seconds.

Flow monitoring on the DUT uses the destination IP address as the Flow Key.

After the first 100 packets are sent, 100 Cache entries will have been created in the Flow monitoring Cache. The subsequent packets will be counted against the already created Cache entries since the destination IP address (Flow Key) has already been seen by the DUT (provided the Cache entries did not expire yet as described below).

A packet with destination IP address equal to A is sent every 0.1 second, so the Cache entry is refreshed in the Cache every 0.1 second, while the Inactive Timeout is 10 seconds. In this case the

Cache entries will not expire until the Active Timeout, e.g. they will expire every 100 seconds and then the Cache entries will be created again.

If the test measurement time is 50 seconds from the start of the traffic generator then the measured Flow Export Rate is 0 since during this period nothing expired from the Cache.

If the test measurement time is 100 seconds from the start of the traffic generator then the measured Flow Export Rate is 1 Flow Record per second.

If the test measurement time is 290 seconds from the start of the traffic generator then the measured Flow Export Rate is 2/3 of Flow Record per second since during the 290 seconds period we expired the same 100 of Flows twice.

5. Flow Monitoring Throughput Measurement Methodology

Objective:

To measure the Flow monitoring performance in a manner comparable between different Flow monitoring implementations.

Metric definition:

Flow Monitoring Throughput - see section 3.

Discussion:

Different Flow monitoring implementations might chose to handle Flow Export from a partially empty Cache differently than in the case when the Cache fully occupied. Similarly software and hardware based DUTs can handle the same situation as stated above differently. The purpose of the benchmark measurement in this section is to abstract from all the possible behaviours and define one measurement procedure covering all the possibilities. The only criteria is to measure as defined here until Flow Record or packet losses are seen. The decision whether to dive deeper into the conditions under which the packet losses happen is left to the tester.

5.1 Flow Monitoring Configuration

Cache Size

Cache Size configuration is dictated by the expected position of the DUT in the network and by the chosen Flow Keys of the Flow Record. The number of unique Flow Keys sets that the traffic generator (sender) provides should be multiple times larger than the Cache Size, to ensure that the existing Cache entries are never updated before Flow Expiration and Flow Export. The Cache Size MUST be known in order to define the measurement circumstances properly.

Inactive Timeout

Inactive Timeout is set (if configurable) to the minimum possible value on the DUT. This ensures that the Cache entries are expired as soon as possible and exported out of the DUT Cache. It MUST be known in order to define the measurement circumstances completely and equally across implementations.

Active Timeout

Active Timeout is set (if configurable) to a value equal to or higher than the Inactive Timeout. It MUST be known in order to define the measurement circumstances completely and equally across implementations.

Flow Keys Definition:

The test needs large numbers of unique Cache entries to be created by incrementing values of one or several Flow Keys. The number of unique combinations of Flow Keys values SHOULD be several times larger than the DUT Cache Size. This makes sure that any incoming packet will never refresh any already existing Cache entry.

The availability of Cache Size, Inactive Timeout, Active Timeout as configuration parameters is implementation specific. If the Flow monitoring implementation does not support it, the test possibilities as specified by this document are restricted. Some testing might be viable if the implementation follows the [IPFIX-CONFIG] document and needs to be considered on the case by case basis.

5.2 Traffic Configuration

Traffic Generation

The traffic generator needs to increment the Flow Keys values with each sent packet, this way each packet represents one Cache entry in the DUT Cache.

If the test traffic rate is below the maximum media rate for the particular packet size the traffic generator MUST send the packets in equidistant time intervals. Traffic generators which do not fulfil this condition MUST NOT and cannot be used for the Flow Monitoring Throughput measurement. An example of this behaviour is if the test traffic rate is one half of the media rate and the traffic generator achieves this by sending each half of the second at the full media rate and then sending nothing for the second half of the second. In such conditions it would be impossible to distinguish if the DUT failed to handle the Flows due to the input buffers shortage during the burst or due to the limits in the Flow Monitoring performance.

Measurement Duration

The measurement duration (e.g. how long the test traffic is sent to the DUT) MUST be at least two times longer than the Inactive Timeout otherwise no Flow Export would be seen. The measurement duration SHOULD guarantee that the number of Cache entries created during the measurement exceeds the available Cache Size.

5.3 Cache Population

The product of Inactive Timeout and the packet rate offered to the DUT (cache population) during the measurements determines the total number of Cache entries in the DUT Cache during one particular measurement (while taking into account some margin for dynamic

behaviour during high DUT loads when processing the Flows).

Novak

Expires April, 2012

[Page 19]

Internet-Draft

Flow Monitoring Benchmarking

October 2011

The Flow monitoring implementation might behave differently depending on the relation of cache population to the available Cache Size during the measurement. This behaviour is fully implementation specific and will also be influenced if the DUT is software based or hardware based architecture.

The cache population (if it is lower or higher than the available Cache Size) during a particular benchmark measurement SHOULD be noted and mainly only measurements with same cache population SHOULD be compared.

5.4 Measurement Time Interval

The measurement time interval is the time value which is used to calculate the measured Flow Export Rate from the captured Flow Export data. It is obtained as specified below.

RFC2544 specifies with the precision of the packet beginning and end the time intervals to be used to measure the DUT time characteristics. In the case of a Flow Monitoring Throughput measurement the start and stop time needs to be clearly defined but the granularity of this definition can be limited to just marking the start and stop time with the start and stop of the traffic generator. This assumes that the traffic generator and DUT are collocated and the variance in transmission delay from the generator to the DUT is negligible as compared to the total time of traffic generation.

The measurement start time: the time when the traffic generator is started

The measurement stop time: the time when the traffic generator is stopped

The measurement time interval is then calculated as the difference (stop time) - (start time) - (Inactive Timeout).

This supposes that the Cache Size is large enough so that the time to fill it up with Cache entries is longer than Inactive Timeout. Otherwise the time to fill up the Cache needs to be used for calculation of the measurement time interval in the place of the Inactive Timeout.

Instead of measuring the absolute values of stop and start time it is possible to setup the traffic generator to send traffic for a certain pre-defined time interval which is then used in the above definition instead of the difference (stop time) - (start time).

The Collector MUST stop collecting the Flow Export data at the measurement stop time.

The Inactive Timeout (or the time needed to fill up the Cache) causes delay of the Flow Export data behind the test traffic which is analysed by the DUT. E.g. if the traffic starts at time point X Flow

Export will start only at the time point $X + \text{Inactive Timeout}$ (or $X +$ time to fill up the Cache). Since Flow Export capture needs to stop with the traffic (because that's when the DUT stops processing the Flows at the given rate) the time interval during which the DUT kept exporting data is shorter by the Inactive Timeout than the Time interval when the test traffic was sent from the traffic generator to the DUT.

5.5 Flow Export Rate Measurement

The Flow Export Rate needs to be measured in two consequent steps. The purpose of the first step (point a. below) is to gain the actual value for the rate, the second step (point b. below) needs to be done in order to verify Flow Record drops during the measurement:

- a. In the first step the captured Flow Export data MUST be analyzed only for the capturing interval (measurement time interval) as specified in section 5.4. During this period the DUT is forced to process Cache entries at the rate the packets are sent. When traffic generation finishes, the behaviour when emptying the Cache is completely implementation specific and the Flow Export data from this period cannot be therefore used for the benchmarking.
- b. In the second step all the Flow Export data from the DUT MUST be captured in order to be capable to determine the Flow Record losses. It needs to be taken into account that especially when large Cache Sizes (in order of magnitude of hundreds of thousands of entries and higher) are in use the Flow Export can take many multiples of Inactive Timeout to empty the Cache after the measurement. This behaviour is completely implementation specific.

If the Collector has the capability to redirect the Flow Export data after the measurement time interval into different capture buffer (or time stamp the received Flow Export data after that) this can be done in one step. Otherwise each Flow Monitoring Throughput measurement at certain packet rate needs to be executed twice - once to capture the Flow Export data just for the measurement time interval (to determine the actual Flow Export Rate) and second time to capture all Flow Export data in order to determine Flow Record losses at that packet rate.

At the end of the measurement time interval the DUT might still be processing Cache entries which belong to the Flows expired from the Cache before the end of the interval while they will appear in an export packet sent only after the end of the measurement interval. This imprecision can be mitigated by large amounts of Flow Records used during the measurement (so that the few Flow Records in one export packet can be ignored) or by use of timestamps exported with the Flow Records.

5.6 The Measurement Procedure

The measurement procedure is same as the Throughput measurement in section 26.1 of [RFC2544] for the traffic sending side. The DUT

output analysis is done on the traffic generator receiving side for the test traffic the same way as for RFC2544 measurements.

An additional analysis is performed using data captured by the Collector. The purpose of this analysis is to establish the value of the Flow Export Rate during the current measurement step and to verify

that no Flow Records were dropped during the measurement. The procedure to measure Flow Export Rate is described in section 5.5.

The Flow Export performance can be significantly affected by the way the Flow monitoring implementation formats the Flow Records into the Flow Export packets in terms of ordering and frequency of Control Information export and mainly the number of Flow Records in one Flow Export packet. The worst case scenario here is just one Flow Record in every Flow Export packet.

Flow Export data should be sanity checked during the benchmark measurement for:

- a. the number of Flow Records per packet, by simply calculating the ratio of exported Flow Records to the number of Flow Export packets captured during the measurement (which should be available as a counter on the Collector capture buffer)
- b. the number Flow Records corresponding to the export of Control Information per Flow Export packet (calculated as the ratio of the total number of such Flow Records in the Flow Export data and the number of Flow Export packets).

6. RFC2544 Measurements

RFC2544 measurements can be performed under two Flow Monitoring set-ups (see also section 3.4.2). This section details both of them and specifies ways to construct the test traffic so that RFC2544 measurements can be performed in a controlled environment from the Flow monitoring point of view. A controlled Flow monitoring environment means that the tester always knows what Flow monitoring activity (Flow Export Rate) the traffic offered to the DUT causes.

This section is applicable mainly for the RFC2544 throughput (RFC2544 section 26.1) and latency (RFC2544 section 26.2) measurements. It could be used also to measure frame loss rate (RFC2544 section 26.3) and back-to-back frames (RFC2544 section 26.4). It is not relevant for the rest of RFC2544 network interconnect devices characteristics.

Objective:

Provide RFC2544 network device characteristics in the presence of Flow monitoring on the DUT. RFC2544 studies numerous characteristics of network devices. The DUT forwarding and time characteristics without Flow monitoring present on the DUT can vary significantly when Flow monitoring is deployed on the network device.

Metric definition:

Metric as specified in [RFC2544].

The measured RFC2544 Throughput MUST NOT include the packet rate corresponding to the Flow Export data, because it is control type traffic, generated by the DUT as a result of enabling Flow monitoring and does not contribute to the test traffic which the DUT can handle. It requires DUT resources to be generated and transmitted and therefore the RFC2544 Throughput in most cases will be much lower when Flow monitoring is enabled on the DUT than without it.

6.1 Flow Monitoring Configuration

Flow monitoring configuration (as detailed in section 4.3) needs to be applied the same way as discussed in section 5 with the exception of the Active Timeout configuration.

The Active Timeout SHOULD be configured to exceed several times the measurement time interval (see section 5.4). This makes sure that if measurements with two traffic components are performed (see section 6.5) there is no Flow monitoring activity related to the second traffic component.

The Flow monitoring configuration does not change in any other way for the measurement performed in this section. What changes and makes the difference is the traffic configurations as specified in the sections below.

6.2 Measurements with the Flow Monitoring Throughput Set-up

The major requirement to perform a measurement with Flow Monitoring Throughput set-up is that the traffic and Flow monitoring is configured in such a way that each sent packet creates one entry in the DUT Cache. This restricts the possible set-ups only to the measurement with two traffic components as specified in section 6.5.

6.3 Measurements With Fixed Flow Export Rate

This section covers the measurements where the RFC2544 metrics need to be measured with Flow monitoring enabled but at certain Flow Export Rate lower than Flow Monitoring Throughput.

The tester here has both options as specified in section 6.4 and 6.5.

6.4 Measurements With Single Traffic Component

Section 12 of [RFC2544] discusses the use of protocol source and destination addresses for defined measurements. To perform all the RFC2544 type measurements with Flow monitoring enabled the defined

Flow Keys SHOULD contain IP source and destination address. The RFC2544 type measurements with Flow monitoring enabled then can be executed under these additional conditions:

- a. the test traffic is not limited to single unique pair of source and destination addresses
- b. the traffic generator defines test traffic as follows:
allow for a parameter to send N (where N is an integer number starting at 1 and incremented in small steps) packets with source IP address A and destination IP address B before changing both IP addresses to the next value

This test traffic definition allows execution of the Flow monitoring measurements with fixed Flow Export Rate while measuring the DUT RFC2544 characteristics. This set-up is the better option since it best simulates the live network traffic scenario with Flows containing more than just one packet.

The initial packet rate at N equal to 1 defines the Flow Export Rate for the whole measurement procedure. Subsequent increases of N will not change the Flow Export Rate as the time and Cache characteristics of the test traffic stay the same. This set-up is suitable for measurements with Flow Export Rates below the Flow Monitoring Throughput.

6.5 Measurements With Two Traffic Components

The test traffic set-up in section 6.4 might be difficult to achieve with commercial traffic generators or the granularity of the traffic rates as defined by the initial packet rate at N equal to 1 might not be suitable for the required measurement. An alternate mechanism is to define two traffic components in the test traffic. One to populate Flow monitoring Cache and the second one to execute the RFC2544 measurements.

- a. Flow monitoring test traffic component - the exact traffic definition as specified in section 5.2.
- b. RFC2544 Test Traffic Component - test traffic as specified by RFC2544 MUST create just one entry in the DUT Cache. In the particular set-up discussed here this would mean a traffic stream with just one pair of unique source and destination IP addresses (but could be avoided if Flow Keys were for example UDP/TCP source and destination ports and Flow Keys did not contain the addresses).

The Flow monitoring traffic component will exercise the DUT in terms of Flow activity while the second traffic component will measure the RFC2544 characteristics.

The measured RFC2544 Throughput is the sum of the packet rates of both traffic components. The definition of other RFC2544 metrics remains unchanged.

7. Flow Monitoring Accuracy

The pure Flow Monitoring Throughput measurement in section 5 provides the capability to verify the Flow monitoring accuracy in terms of the exported Flow Record data. Since every Cache entry created in the Cache is populated by just one packet, the full set of captured data on the Collector can be parsed (e.g. providing the values of all Flow Keys and other Flow Record fields, not only the overall Flow Record count in the exported data) and each set of parameters from each Flow Record can be checked against the parameters as configured on the traffic generator and set in packets sent to the DUT. The exported Flow Record is considered accurate if:

- a. all the Flow Record fields are present in each exported Flow Record
- b. all the Flow Record fields values match the value ranges as set by the traffic generator (for example an IP address falls within the range of the IP addresses increments on the traffic generator)
- c. all the possible Flow Record fields values as defined at the traffic generator have been found in the captured export data on the Collector. This check needs to be offset against detected packet losses at the DUT during the measurement

8. Evaluating Flow Monitoring Applicability

The measurement results as discussed in this document and obtained for certain DUTs allow for a preliminary analysis of a Flow monitoring deployment based on the traffic analysis data from the providers network.

An example of such traffic analysis in the Internet is provided by [CAIDA] and the way it can be used is discussed below. The data needed to make an estimate if a certain network device can manage the particular amount of live traffic with Flow monitoring enabled is:

Average packet size: 350 bytes
Number of packets per IP Flow: 20

Expected data rate on the network device: 1 Gbit/s

The required value needed to be known is the average number of Flows created per second in the network device:

Flows per second =
$$\frac{\text{Expected packet rate}}{\text{Packet per flow}}$$

When using the example values given above, the network device would Be required to process 18 000 Flows per second. By executing the benchmarking as specified in this document a platform capable of this processing can be determined for the deployment in that particular part of the user network.

It needs to be kept in mind that the above is a very rough and averaged Flow activity estimate which cannot account for traffic anomalies, for example a large number of DNS request packets which are typically small packets coming from many different sources and represent mostly just one packet per Flow.

9. Acknowledgements

This work could have been performed thanks to the patience and support of Cisco Systems NetFlow development team, namely Paul Aitken, Paul Atkins and Andrew Johnson. Thanks belong to Benoit Claise for numerous detailed reviews and presentations of the document and Aamer Akhter for initiating this work. A special acknowledgment needs to go to the whole of the working group and especially to the chair Al Morton for the support and work on this draft and to Paul Aitken for a very detailed technical review.

10. Security Considerations

Documents of this type do not directly affect the security of the Internet or corporate networks as long as benchmarking is not performed on devices or systems connected to operating networks.

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the constraints specified in sections above.

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT.

Special capabilities SHOULD NOT exist in the DUT specifically for benchmarking purposes. Any implications for network security arising from the DUT SHOULD be identical in the lab and in production networks.

11. IANA Considerations

This memo makes no requests of the IANA.

12. References

12.1. Normative References

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

- [RFC2544] Bradner, S., "Benchmarking Methodology for Network Interconnect Devices", Informational, RFC 2544, April 1999

12.2. Informative References

- [RFC1242] Bradner, S., "Benchmarking Terminology for Network Interconnection Devices", RFC 1242, July 1991
- [RFC2285] Mandeville R., "Benchmarking Terminology for LAN Switching Devices", Informational, RFC 2285, November 1998
- [RFC3031] E. Rosen, A. Viswanathan, R. Callon, "Multiprotocol Label Switching Architecture", Standards Track, RFC 3031, January 2001
- [RFC3917] Quittek J., "Requirements for IP Flow Information Export (IPFIX)", Informational, RFC 3917, October 2004.
- [RFC5101] Claise B., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", Standards Track, RFC 5101, January 2008
- [RFC5180] C. Popoviciu, A. Hamza, D. Dugatkin, G. Van de Velde, "IPv6 Benchmarking Methodology for Network Interconnect Devices", Informational, RFC 5180, May 2008
- [RFC5470] Sadasivan, G., Brownlee, N., Claise, B., and J. Quittek, "Architecture Model for IP Flow Information Export", RFC 5470, October 2011
- [RFC5695] Akhter A. "MPLS Forwarding Benchmarking Methodology", RFC 5695, November 2009
- [CAIDA] Claffy, K., "The nature of the beast: recent traffic measurements from an Internet backbone", <http://www.caida.org/publications/papers/1998/Inet98/Inet98.html>
- [IPFIX-CONFIG] Configuration Data Model for IPFIX and PSAMP, G. Muenz et al, Work in Progress, draft-ietf-ipfix-configuration-model-10

Author's Addresses

Jan Novak (editor)
Cisco Systems
Edinburgh,
United Kingdom
Email: janovak@cisco.com

Appendix A: Recommended Report Format

Parameter	Units
Test Case	test case name (section 5 and 6)
Test Topology	Figure 2, other
Traffic Type	IPv4, IPv6, MPLS, other
Test Results	
Flow Monitoring Throughput	Flow Records per second or Not Applicable
Flow Export Rate	Flow Records per second or Not Applicable
Control Information Export Rate	Flow Records per second
RFC2544 Throughput	packets per second
(Other RFC2544 Metrics)	(as appropriate)
General Parameters	
Traffic Direction	unidirectional, bidirectional
DUT Interface Type	Ethernet, POS, ATM, other
DUT Interface Bandwidth	MegaBits per second
Traffic Specifications	
Number of Traffic Components	(see section 6.4 and 6.5)
For each traffic component:	
Packet Size	bytes
Traffic Packet Rate	packets per second
Traffic Bit Rate	MegaBits per second
Number of Packets Sent	number of entries
Incremented Packet Header Fields	list of fields
Number of Unique Header Values	number of entries
Number of Packets per Flow	number of entries
Flow monitoring Specifications	
Direction	ingress, egress, both
Observation Points	DUT interface names
Cache Size	number of entries
Active Timeout	seconds
Inactive Timeout	seconds
Flow Keys	list of fields
Flow Record Fields	total number of fields
Number of Flows Created	number of entries
Flow Export Transport Protocol	UDP, TCP, SCTP, other
Flow Export Protocol	IPFIX, NetFlow, other
Flow Export data packet size	bytes
MPLS Specifications	(for traffic type MPLS only)
Tested Label Operation	imposition, swap, disposition

Appendix B: Miscellaneous Tests

This section lists the tests which could be useful to assess a proper Flow monitoring operation under various operational or stress conditions. These tests are not deemed suitable for any benchmarking for various reasons.

B.1 DUT Under Traffic Load

The Flow Monitoring Throughput SHOULD be measured under different levels of static traffic load through the DUT. This can be achieved only by using two traffic components as discussed in the section 6.5, where one traffic component exercises the Flow Monitoring Plane and the second traffic component loads only the Forwarding Plane without affecting Flow monitoring (e.g. it creates just a certain amount of permanent Cache entries).

The variance in Flow Monitoring Throughput as function of the traffic load should be noted for comparison purposes between two DUTs of similar architecture and capability.

B.2 In-band Flow Export

The test topology in section 4.1 mandates the use of separate Flow Export interface to avoid the Flow Export data generated by the DUT to mix with the test traffic from the traffic generator. This is necessary in order to create clear and reproducible test conditions for the benchmark measurement.

The real network deployment of Flow monitoring might not allow for such a luxury - for example on a very geographically large network. In such a case, Flow Export will use an ordinary traffic forwarding interface e.g. in-band Flow Export.

The Flow monitoring operation should be verified with in-band Flow Export configuration while following these test steps:

- a. Perform benchmark test as specified in section 5
- b. One of the results will be how much bandwidth Flow Export used on the dedicated Flow Export interface
- c. Change Flow Export configuration to use the test interface
- d. Repeat the benchmark test while the receiver filters out the Flow Export data from analysis

The expected result is that the RFC2544 Throughput achieved in step a. is same as the Throughput achieved in step d. provided that the bandwidth of the output DUT interface is not the bottleneck (in other words it must have enough capacity to forward both test and Flow Export traffic).

B.3 Variable Packet Size

The Flow monitoring measurements specified in this document would be interesting to repeat with variable packet sizes within one

particular test (e.g. test traffic containing mix of packet sizes). The packet forwarding tests specified mainly in [RFC2544] do not recommend and perform such tests. Flow monitoring is not dependent on packet sizes so such a test could be performed during the Flow Monitoring Throughput measurement and verify its value does not depend on the offered traffic packet sizes. The tests must be carefully designed in order to avoid measurement errors due to the physical bandwidth limitations and changes of the base forwarding performance with packet size.

B.4 Bursty Traffic

RFC2544 section 21 discusses and defines the use of bursty traffic. It can be used for Flow monitoring testing as well to gauge some short term overload DUT capabilities in terms of Flow monitoring. The tests benchmark here would not be the Flow Export Rate the DUT can sustain but the absolute number of Flow Records the DUT can process without dropping any single Flow Record. The traffic set-up to be used for this test is as follows:

- a. each sent packet creates a new Cache entry
- b. the packet rate is set to the maximum transmission speed of the DUT interface used for the test

B.5 Various Flow Monitoring Configurations

This section translates the terminology used in the IPFIX documents [RFC5470], [RFC5101] and others into the terminology used in this document. Section B.5.2 proposes another measurement which is not possible to verify in a black box test manner.

B.5.1 RFC2544 Throughput without Metering Process

If Metering Process is not defined on the DUT it means no Flow monitoring Cache exists and no Flow analysis occurs. The performance measurement of the DUT in such a case is just pure [RFC2544] measurement.

B.5.2 RFC2544 Throughput with Metering Process

If only Metering Process is enabled it means that Flow analysis on the DUT is enabled and operational but no Flow Export happens. The performance measurement of a DUT in such a configuration represents an useful test of the DUT capabilities (this corresponds to the case when the network operator uses Flow monitoring for example for manual denial of service attacks detection and does not wish to use Flow Export).

The performance testing on this DUT can be performed as discussed in this document but it is not possible to verify the operation and results without interrogating the DUT.

B.5.3 RFC2544 Throughput with Metering and Exporting Process

This test represents the performance testing as discussed in section 6.

B.6 Tests With Bidirectional Traffic

The test topology on figure 2 can be expanded to verify Flow monitoring functionality with bidirectional traffic in two possible ways:

- a. use two sets of interfaces, one for Flow monitoring for ingress traffic and one for Flow monitoring egress traffic
- b. use exactly same set-up as in figure 2 but use the interfaces in full duplex mode e.g. sending and receiving simultaneously on each of them

The set-up in point a. above is in fact equivalent to the set-up with several Observation Points as already discussed in section 4.1 and 4.3.1.

For the set-up in point b. same rules should be applied (as per section 4.1 and 4.3.1) - traffic passing through each Observation Point SHOULD always create a new Cache entry in the Cache e.g. the same traffic SHOULD NOT be just looped back on the receiving interfaces to create the bidirectional traffic flow.

B.7 Instantaneous Flow Export Rate

An additional useful information when analysing the Flow Export data is the time distribution of the instantaneous Flow Export Rate. It can be derived during the measurements in two ways:

- a. The Collector might provide the capability to decode Flow Export during capturing and at the same time counting the Flow Records and provide the instantaneous (or simply an average over shorter time interval than specified in section 5.4) Flow Export Rate
- b. The Flow Export protocol (like IPFIX [RFC5101]) can provide time stamps in the Flow Export packets which would allow time based analysis and calculate the Flow Export Rate as an average over much shorter time interval than specified in section 5.4

The accuracy and shortest time average will always be limited by the precision of the time stamps (1 second for IPFIX) or by the capabilities of the DUT and the Collector.

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 28, 2012

R. Papneja
Huawei Technologies
S. Vapiwala
J. Karthik
Cisco Systems
S. Poretsky
Allot Communications
S. Rao
Qwest Communications
J. Roux
France Telecom
October 26, 2011

Methodology for benchmarking MPLS protection mechanisms
draft-ietf-bmwg-protection-meth-09.txt

Abstract

This draft describes the methodology for benchmarking MPLS Protection mechanisms for link and node protection as defined in [MPLS-FRR-EXT]. This document provides test methodologies and testbed setup for measuring failover times while considering all dependencies that might impact faster recovery of real-time applications bound to MPLS based traffic engineered tunnels. The benchmarking terms used in this document are defined in [TERM-ID].

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 <http://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 April 28, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1. Introduction	5
2. Document Scope	6
3. Existing Definitions and Requirements	6
4. General Reference Topology	7
5. Test Considerations	8
5.1. Failover Events [TERM-ID]	8
5.2. Failure Detection [TERM-ID]	9
5.3. Use of Data Traffic for MPLS Protection benchmarking	9
5.4. LSP and Route Scaling	10
5.5. Selection of IGP	10
5.6. Restoration and Reversion [TERM-ID]	10
5.7. Offered Load	11
5.8. Tester Capabilities	11
6. Reference Test Setup	12
6.1. Link Protection	12
6.1.1. Link Protection - 1 hop primary (from PLR) and 1 hop backup TE tunnels	12
6.1.2. Link Protection - 1 hop primary (from PLR) and 2 hop backup TE tunnels	13
6.1.3. Link Protection - 2+ hop (from PLR) primary and 1 hop backup TE tunnels	13
6.1.4. Link Protection - 2+ hop (from PLR) primary and 2 hop backup TE tunnels	14
6.2. Node Protection	14
6.2.1. Node Protection - 2 hop primary (from PLR) and 1 hop backup TE tunnels	14
6.2.2. Node Protection - 2 hop primary (from PLR) and 2 hop backup TE tunnels	15
6.2.3. Node Protection - 3+ hop primary (from PLR) and 1 hop backup TE tunnels	16
6.2.4. Node Protection - 3+ hop primary (from PLR) and 2 hop backup TE tunnels	17
7. Test Methodology	17
7.1. MPLS FRR Forwarding Performance	18
7.1.1. Headend PLR Forwarding Performance	18
7.1.2. Mid-Point PLR Forwarding Performance	19
7.1.3. Egress PLR Forwarding Performance	20
7.2. Headend PLR with Link Failure	21
7.3. Mid-Point PLR with Link Failure	23
7.4. Headend PLR with Node Failure	24
7.5. Mid-Point PLR with Node Failure	26
8. Reporting Format	27
9. Security Considerations	29
10. IANA Considerations	29
11. Acknowledgements	30
12. References	30

12.1. Informative References	30
12.2. Normative References	31
Appendix A. Fast Reroute Scalability Table	31
Appendix B. Abbreviations	33
Authors' Addresses	33

1. Introduction

This draft describes the methodology for benchmarking MPLS based protection mechanisms. The new terminology that this document introduces is defined in [TERM-ID].

MPLS based protection mechanisms provide fast recovery of real-time services from a planned or an unplanned link or node failures. MPLS protection mechanisms are generally deployed in a network infrastructure where MPLS is used for provisioning of point-to-point traffic engineered tunnels (tunnel). MPLS based protection mechanisms promise to improve service disruption period by minimizing recovery time from most common failures.

Network elements from different manufacturers behave differently to network failures, which impacts the network's ability and performance for failure recovery. It therefore becomes imperative for service providers to have a common benchmark to understand the performance behaviors of network elements.

There are two factors impacting service availability: frequency of failures and duration for which the failures persist. Failures can be classified further into two types: correlated and uncorrelated. Correlated and uncorrelated failures may be planned or unplanned.

Planned failures are predictable. Network implementations should be able to handle both planned and unplanned failures and recover gracefully within a time frame to maintain service assurance. Hence, failover recovery time is one of the most important benchmark that a service provider considers in choosing the building blocks for their network infrastructure.

A correlated failure is the simultaneous occurrence of two or more failures. A typical example is failure of a logical resource (e.g. layer-2 links) due to a dependency on a common physical resource (e.g. common conduit) that fails. Within the context of MPLS protection mechanisms, failures that arise due to Shared Risk Link Groups (SRLG) [MPLS-FRR-EXT] can be considered as correlated failures. Not all correlated failures are predictable in advance, for example, those caused by natural disasters.

MPLS Fast Re-Route (MPLS-FRR) allows for the possibility that the Label Switched Paths can be re-optimized in the minutes following Failover. IP Traffic would be re-routed according to the preferred path for the post-failure topology. Thus, MPLS-FRR includes an additional step to the General model:

- (1) Failover Event - Primary Path (Working Path) fails
- (2) Failure Detection- Failover Event is detected
- (3)
 - a. Failover - Working Path switched to Backup path
 - b. Re-Optimization of Working Path (possible change from Backup Path)
- (4) Restoration - Primary Path recovers from a Failover Event
- (5) Reversion (optional) - Working Path returns to Primary Path

2. Document Scope

This document provides detailed test cases along with different topologies and scenarios that should be considered to effectively benchmark MPLS protection mechanisms and failover times on the Data Plane. Different Failover Events and scaling considerations are also provided in this document.

All benchmarking testcases defined in this document apply to both facility backup and local protection enabled in detour mode. The test cases cover all possible failure scenarios and the associated procedures benchmark the performance of the Device Under Test (DUT) to recover from failures. Data plane traffic is used to benchmark failover times.

Benchmarking of correlated failures is out of scope of this document. Protection from Bi-directional Forwarding Detection (BFD) is outside the scope of this document.

As described above, MPLS-FRR may include a Re-optimization of the Working Path, with possible packet transfer impairments. Characterization of Re-optimization is beyond the scope of this memo.

3. Existing Definitions and Requirements

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 BCP 14, RFC 2119 [Br97]. RFC 2119 defines the use of these key words to help make the

intent of standards track documents as clear as possible. While this document uses these keywords, this document is not a standards track document.

The reader is assumed to be familiar with the commonly used MPLS terminology, some of which is defined in [MPLS-FRR-EXT].

This document uses much of the terminology defined in [TERM-ID]. This document also uses existing terminology defined in other BMWG work. Examples include, but are not limited to:

Throughput	[Ref.[Br91], section 3.17]
Device Under Test (DUT)	[Ref.[Ma98], section 3.1.1]
System Under Test (SUT)	[Ref.[Ma98], section 3.1.2]
Out-of-order Packet	[Ref.[Po06], section 3.3.2]
Duplicate Packet	[Ref.[Po06], section 3.3.3]

4. General Reference Topology

Figure 1 illustrates the basic reference testbed and is applicable to all the test cases defined in this document. The Tester is comprised of a Traffic Generator (TG) & Test Analyzer (TA). A Tester is directly connected to the DUT. The Tester sends and receives IP traffic to the tunnel ingress and performs signaling protocol emulation to simulate real network scenarios in a lab environment. The Tester may also support MPLS-TE signaling to act as the ingress node to the MPLS tunnel.

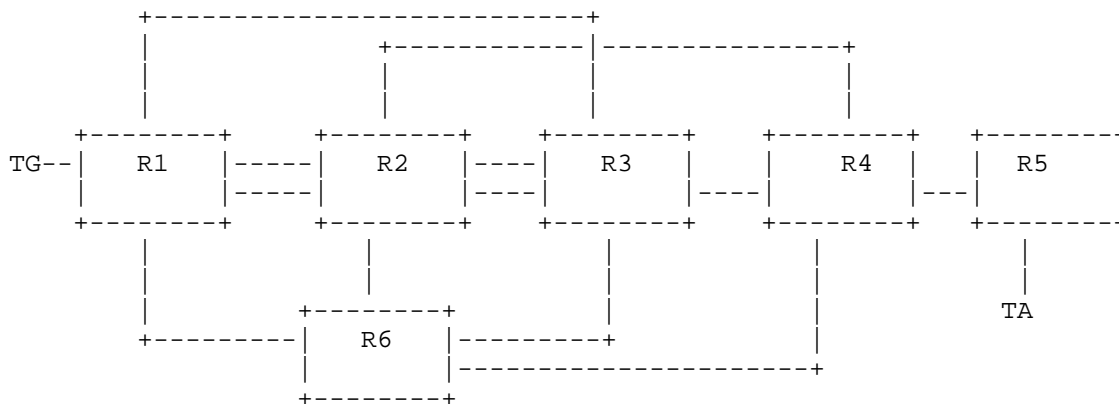


Fig. 1 Fast Reroute Topology

The tester MUST record the number of lost, duplicate, and reordered packets. It should further record arrival and departure times so that Failover Time, Additive Latency, and Reversion Time can be measured. The tester may be a single device or a test system emulating all the different roles along a primary or backup path.

The label stack is dependent of the following 3 entities:

- (1) Type of protection (Link Vs Node)
- (2) # of remaining hops of the primary tunnel from the PLR
- (3) # of remaining hops of the backup tunnel from the PLR

Due to this dependency, it is RECOMMENDED that the benchmarking of failover times be performed on all the topologies provided in section 6.

5. Test Considerations

This section discusses the fundamentals of MPLS Protection testing:

- (1) The types of network events that causes failover
- (2) Indications for failover
- (3) the use of data traffic
- (4) Traffic generation
- (5) LSP Scaling
- (6) Reversion of LSP
- (7) IGP Selection

5.1. Failover Events [TERM-ID]

The failover to the backup tunnel is primarily triggered by either link or node failures observed downstream of the Point of Local repair (PLR). Some of these failure events are listed below.

Link Failure Events

- Interface Shutdown on PLR side with POS Alarm
- Interface Shutdown on remote side with POS Alarm
- Interface Shutdown on PLR side with RSVP hello enabled
- Interface Shutdown on remote side with RSVP hello enabled
- Interface Shutdown on PLR side with BFD
- Interface Shutdown on remote side with BFD
- Fiber Pull on the PLR side (Both TX & RX or just the TX)
- Fiber Pull on the remote side (Both TX & RX or just the RX)
- Online insertion and removal (OIR) on PLR side
- OIR on remote side
- Sub-interface failure (e.g. shutting down of a VLAN)
- Parent interface shutdown (an interface bearing multiple sub-interfaces)

Node Failure Events

- A System reload initiated either by a graceful shutdown or by a power failure.
- A system crash due to a software failure or an assert.

5.2. Failure Detection [TERM-ID]

Link failure detection time depends on the link type and failure detection protocols running. For SONET/SDH, the alarm type (such as LOS, AIS, or RDI) can be used. Other link types have layer-two alarms, but they may not provide a short enough failure detection time. Ethernet based links do not have layer 2 failure indicators, and therefore relies on layer 3 signaling for failure detection. However for directly connected devices, remote fault indication in the ethernet auto-negotiation scheme could be considered as a type of layer 2 link failure indicator.

MPLS has different failure detection techniques such as BFD, or use of RSVP hellos. These methods can be used for the layer 3 failure indicators required by Ethernet based links, or for some other non-Ethernet based links to help improve failure detection time.

The test procedures in this document can be used for a local failure or remote failure scenarios for comprehensive benchmarking and to evaluate failover performance independent of the failure detection techniques.

5.3. Use of Data Traffic for MPLS Protection benchmarking

Currently end customers use packet loss as a key metric for Failover Time [TERM-ID]. Failover Packet Loss [TERM-ID] is an externally observable event and has direct impact on application performance. MPLS protection is expected to minimize the packet loss in the event

of a failure. For this reason it is important to develop a standard router benchmarking methodology for measuring MPLS protection that uses packet loss as a metric. At a known rate of forwarding, packet loss can be measured and the failover time can be determined. Measurement of control plane signaling to establish backup paths is not enough to verify failover. Failover is best determined when packets are actually traversing the backup path.

An additional benefit of using packet loss for calculation of failover time is that it allows use of a black-box test environment. Data traffic is offered at line-rate to the device under test (DUT) an emulated network failure event is forced to occur, and packet loss is externally measured to calculate the convergence time. This setup is independent of the DUT architecture.

In addition, this methodology considers the packets in error and duplicate packets that could have been generated during the failover process. The methodologies consider lost, out-of-order, and duplicate packets to be impaired packets that contribute to the Failover Time.

5.4. LSP and Route Scaling

Failover time performance may vary with the number of established primary and backup tunnel label switched paths (LSP) and installed routes. However the procedure outlined here should be used for any number of LSPs (L) and number of routes protected by PLR(R). The amount of L and R must be recorded.

5.5. Selection of IGP

The underlying IGP could be ISIS-TE or OSPF-TE for the methodology proposed here. See [IGP-METH] for IGP options to consider and report.

5.6. Restoration and Reversion [TERM-ID]

Fast Reroute provides a method to return or restore an original primary LSP upon recovery from the failure (Restoration) and to switch traffic from the Backup Path to the restored Primary Path (Reversion). In MPLS-FRR, Reversion can be implemented as Global Reversion or Local Reversion. It is important to include Restoration and Reversion as a step in each test case to measure the amount of packet loss, out of order packets, or duplicate packets that is produced.

Note: In addition to restoration and reversion, re-optimization can take place while the failure is still not recovered but it depends on

the user configuration, and re-optimization timers.

5.7. Offered Load

It is suggested that there be one or more traffic streams as long as there is a steady and constant rate of flow for all the streams. In order to monitor the DUT performance for recovery times, a set of route prefixes should be advertised before traffic is sent. The traffic should be configured towards these routes.

At least 16 flows should be used, and more if possible. Prefix-dependency behaviors are key in IP and tests with route-specific flows spread across the routing table will reveal this dependency. Generating traffic to all of the prefixes reachable by the protected tunnel (probably in a Round-Robin fashion, where the traffic is destined to all the prefixes but one prefix at a time in a cyclic manner) is not recommended. The reason why traffic generation is not recommended in a Round-Robin fashion to all the prefixes, one at a time is that if there are many prefixes reachable through the LSP the time interval between 2 packets destined to one prefix may be significantly high and may be comparable with the failover time being measured which does not aid in getting an accurate failover measurement.

5.8. Tester Capabilities

It is RECOMMENDED that the Tester used to execute each test case have the following capabilities:

- 1.Ability to establish MPLS-TE tunnels and push/pop labels.
- 2.Ability to produce Failover Event [TERM-ID].
- 3.Ability to insert a timestamp in each data packet's IP payload.
- 4.An internal time clock to control timestamping, time measurements, and time calculations.
- 5.Ability to disable or tune specific Layer-2 and Layer-3 protocol functions on any interface(s).
- 6.Ability to react upon the receipt of path error from the PLR

The Tester MAY be capable to make non-data plane convergence observations and use those observations for measurements.

Mid-point LSPs 0 0

6.1.2. Link Protection - 1 hop primary (from PLR) and 2 hop backup TE tunnels

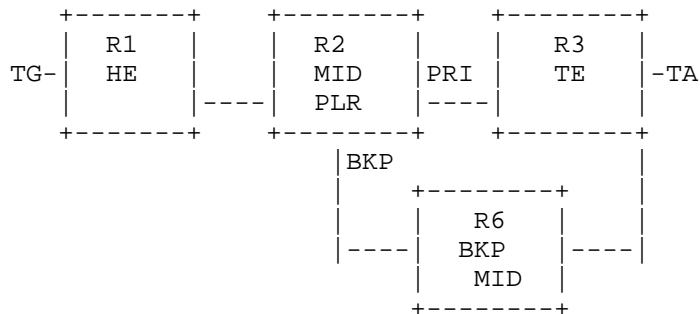


Figure 3.

Traffic	Num of Labels before failure	Num of labels after failure
IP TRAFFIC (P-P)	0	1
Layer3 VPN (PE-PE)	1	2
Layer3 VPN (PE-P)	2	3
Layer2 VC (PE-PE)	1	2
Layer2 VC (PE-P)	2	3
Mid-point LSPs	0	1

6.1.3. Link Protection - 2+ hop (from PLR) primary and 1 hop backup TE tunnels

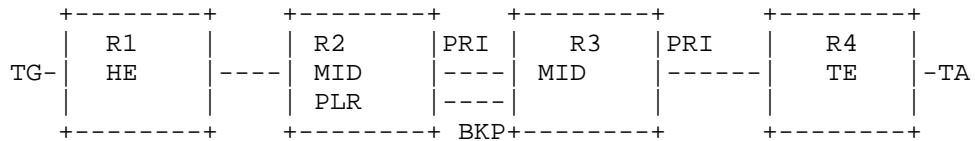


Figure 4.

Traffic	Num of Labels before failure	Num of labels after failure
IP TRAFFIC (P-P)	1	1
Layer3 VPN (PE-PE)	2	2
Layer3 VPN (PE-P)	3	3
Layer2 VC (PE-PE)	2	2
Layer2 VC (PE-P)	3	3
Mid-point LSPs	1	1

6.1.4. Link Protection - 2+ hop (from PLR) primary and 2 hop backup TE tunnels

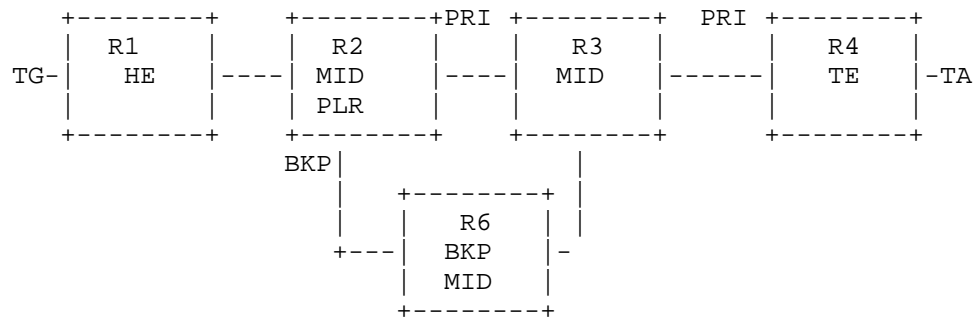


Figure 5.

Traffic	Num of Labels before failure	Num of labels after failure
IP TRAFFIC (P-P)	1	2
Layer3 VPN (PE-PE)	2	3
Layer3 VPN (PE-P)	3	4
Layer2 VC (PE-PE)	2	3
Layer2 VC (PE-P)	3	4
Mid-point LSPs	1	2

6.2. Node Protection

6.2.1. Node Protection - 2 hop primary (from PLR) and 1 hop backup TE tunnels

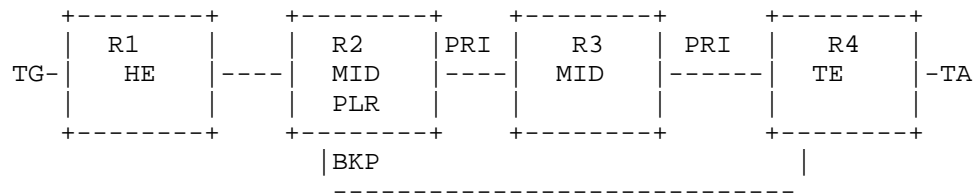


Figure 6.

Traffic	Num of Labels before failure	Num of labels after failure
IP TRAFFIC (P-P)	1	0
Layer3 VPN (PE-PE)	2	1
Layer3 VPN (PE-P)	3	2
Layer2 VC (PE-PE)	2	1
Layer2 VC (PE-P)	3	2
Mid-point LSPs	1	0

6.2.2. Node Protection - 2 hop primary (from PLR) and 2 hop backup TE tunnels

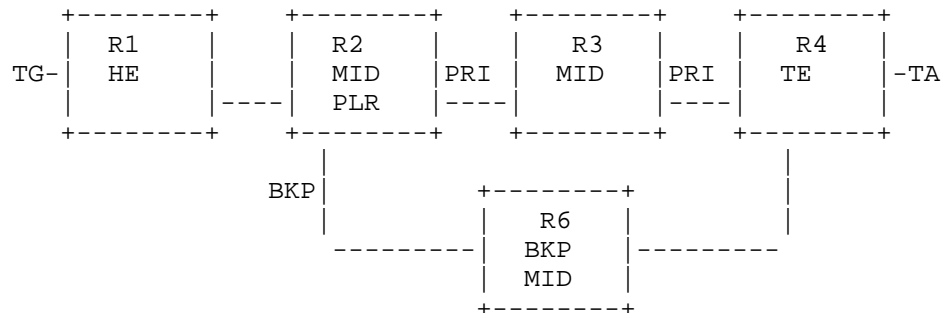


Figure 7.

Traffic	Num of Labels before failure	Num of labels after failure
IP TRAFFIC (P-P)	1	1
Layer3 VPN (PE-PE)	2	2
Layer3 VPN (PE-P)	3	3
Layer2 VC (PE-PE)	2	2
Layer2 VC (PE-P)	3	3
Mid-point LSPs	1	1

6.2.3. Node Protection - 3+ hop primary (from PLR) and 1 hop backup TE tunnels

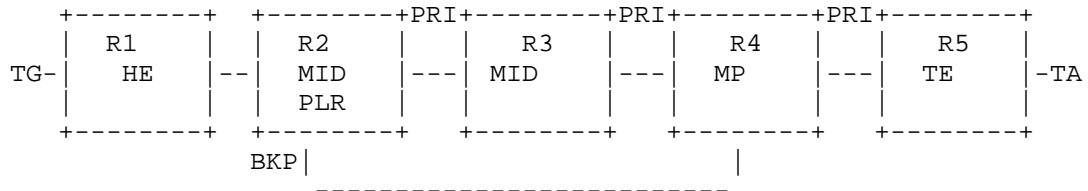


Figure 8.

Traffic	Num of Labels before failure	Num of labels after failure
IP TRAFFIC (P-P)	1	1
Layer3 VPN (PE-PE)	2	2
Layer3 VPN (PE-P)	3	3
Layer2 VC (PE-PE)	2	2
Layer2 VC (PE-P)	3	3
Mid-point LSPs	1	1

6.2.4. Node Protection - 3+ hop primary (from PLR) and 2 hop backup TE tunnels

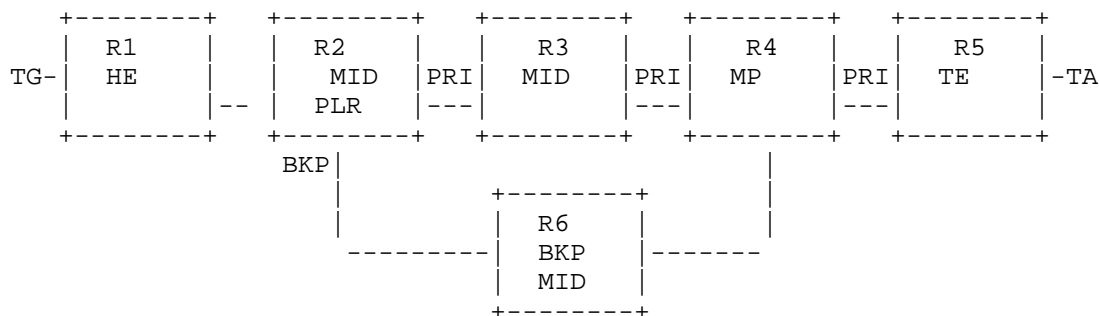


Figure 9.

Traffic	Num of Labels before failure	Num of labels after failure
IP TRAFFIC (P-P)	1	2
Layer3 VPN (PE-PE)	2	3
Layer3 VPN (PE-P)	3	4
Layer2 VC (PE-PE)	2	3
Layer2 VC (PE-P)	3	4
Mid-point LSPs	1	2

7. Test Methodology

The procedure described in this section can be applied to all the 8 base test cases and the associated topologies. The backup as well as the primary tunnels are configured to be alike in terms of bandwidth usage. In order to benchmark failover with all possible label stack depth applicable as seen with current deployments, it is RECOMMENDED to perform all of the test cases provided in this section. The forwarding performance test cases in section 7.1 MUST be performed prior to performing the failover test cases.

The considerations of Section 4 of [RFC2544] are applicable when evaluating the results obtained using these methodologies as well.

7.1. MPLS FRR Forwarding Performance

Benchmarking Failover Time [TERM-ID] for MPLS protection first requires baseline measurement of the forwarding performance of the test topology including the DUT. Forwarding performance is benchmarked by the Throughput as defined in [MPLS-FWD] and measured in units pps. This section provides two test cases to benchmark forwarding performance. These are with the DUT configured as a Headend PLR, Mid-Point PLR, and Egress PLR.

7.1.1. Headend PLR Forwarding Performance

Objective:

To benchmark the maximum rate (pps) on the PLR (as headend) over primary LSP and backup LSP.

Test Setup:

- A. Select any one topology out of the 8 from section 6.
- B. Select overlay technologies (e.g. IGP, VPN, or VC) with DUT as Headend PLR.
- C. The DUT will also have 2 interfaces connected to the traffic Generator/analyzer. (If the node downstream of the PLR is not a simulated node, then the Ingress of the tunnel should have one link connected to the traffic generator and the node downstream to the PLR or the egress of the tunnel should have a link connected to the traffic analyzer).

Procedure:

1. Establish the primary LSP on R2 required by the topology selected.
2. Establish the backup LSP on R2 required by the selected topology.
3. Verify primary and backup LSPs are up and that primary is protected.
4. Verify Fast Reroute protection is enabled and ready.

5. Setup traffic streams as described in section 5.7.
6. Send MPLS traffic over the primary LSP at the Throughput supported by the DUT.
7. Record the Throughput over the primary LSP.
8. Trigger a link failure as described in section 5.1.
9. Verify that the offered load gets mapped to the backup tunnel and measure the Additive Backup Delay.
10. 30 seconds after Failover, stop the offered load and measure the Throughput, Packet Loss, Out-of-Order Packets, and Duplicate Packets over the Backup LSP.
11. Adjust the offered load and repeat steps 6 through 10 until the Throughput values for the primary and backup LSPs are equal.
12. Record the Throughput. This is the offered load that will be used for the Headend PLR failover test cases.

7.1.1.2. Mid-Point PLR Forwarding Performance

Objective:

To benchmark the maximum rate (pps) on the PLR (as mid-point) over primary LSP and backup LSP.

Test Setup:

- A. Select any one topology out of the 9 from section 6.
- B. Select overlay technologies (e.g. IGP, VPN, or VC) with DUT as Mid-Point PLR.
- C. The DUT will also have 2 interfaces connected to the traffic generator.

Procedure:

1. Establish the primary LSP on R1 required by the topology selected.

2. Establish the backup LSP on R2 required by the selected topology.
3. Verify primary and backup LSPs are up and that primary is protected.
4. Verify Fast Reroute protection is enabled and ready.
5. Setup traffic streams as described in section 5.7.
6. Send MPLS traffic over the primary LSP at the Throughput supported by the DUT.
7. Record the Throughput over the primary LSP.
8. Trigger a link failure as described in section 5.1.
9. Verify that the offered load gets mapped to the backup tunnel and measure the Additive Backup Delay.
10. 30 seconds after Failover, stop the offered load and measure the Throughput, Packet Loss, Out-of-Order Packets, and Duplicate Packets over the Backup LSP.
11. Adjust the offered load and repeat steps 6 through 10 until the Throughput values for the primary and backup LSPs are equal.
12. Record the Throughput. This is the offered load that will be used for the Mid-Point PLR failover test cases.

7.1.3. Egress PLR Forwarding Performance

Objective:

To benchmark the maximum rate (pps) on the PLR (as egress) over primary LSP and backup LSP.

Test Setup:

- A. Select any one topology out of the 8 from section 6.
- B. Select overlay technologies (e.g. IGP, VPN, or VC) with DUT as Egress PLR.

- C. The DUT will also have 2 interfaces connected to the traffic generator.

Procedure:

1. Establish the primary LSP on R1 required by the topology selected.
2. Establish the backup LSP on R2 required by the selected topology.
3. Verify primary and backup LSPs are up and that primary is protected.
4. Verify Fast Reroute protection is enabled and ready.
5. Setup traffic streams as described in section 5.7.
6. Send MPLS traffic over the primary LSP at the Throughput supported by the DUT.
7. Record the Throughput over the primary LSP.
8. Trigger a link failure as described in section 5.1.
9. Verify that the offered load gets mapped to the backup tunnel and measure the Additive Backup Delay.
10. 30 seconds after Failover, stop the offered load and measure the Throughput, Packet Loss, Out-of-Order Packets, and Duplicate Packets over the Backup LSP.
11. Adjust the offered load and repeat steps 6 through 10 until the Throughput values for the primary and backup LSPs are equal.
12. Record the Throughput. This is the offered load that will be used for the Egress PLR failover test cases.

7.2. Headend PLR with Link Failure

Objective:

To benchmark the MPLS failover time due to link failure events described in section 5.1 experienced by the DUT which is the Headend PLR.

Test Setup:

- A. Select any one topology out of the 8 from section 6.
- B. Select overlay technology for FRR test (e.g. IGP, VPN, or VC).
- C. The DUT will also have 2 interfaces connected to the traffic Generator/analyzer. (If the node downstream of the PLR is not a simulated node, then the Ingress of the tunnel should have one link connected to the traffic generator and the node downstream to the PLR or the egress of the tunnel should have a link connected to the traffic analyzer).

Test Configuration:

1. Configure the number of primaries on R2 and the backups on R2 as required by the topology selected.
2. Configure the test setup to support Reversion.
3. Advertise prefixes (as per FRR Scalability Table described in Appendix A) by the tail end.

Procedure:

Test Case "7.1.1. Headend PLR Forwarding Performance" MUST be completed first to obtain the Throughput to use as the offered load.

1. Establish the primary LSP on R2 required by the topology selected.
2. Establish the backup LSP on R2 required by the selected topology.
3. Verify primary and backup LSPs are up and that primary is protected.
4. Verify Fast Reroute protection is enabled and ready.

5. Setup traffic streams for the offered load as described in section 5.7.
6. Provide the offered load from the tester at the Throughput [Br91] level obtained from test case 7.1.1.
7. Verify traffic is switched over Primary LSP without packet loss.
8. Trigger a link failure as described in section 5.1.
9. Verify that the offered load gets mapped to the backup tunnel and measure the Additive Backup Delay.
10. 30 seconds after Failover [TERM-ID], stop the offered load and measure the total Failover Packet Loss [TERM-ID].
11. Calculate the Failover Time [TERM-ID] benchmark using the selected Failover Time Calculation Method (TBLM, PLBM, or TBM) [TERM-ID].
12. Restart the offered load and restore the primary LSP to verify Reversion [TERM-ID] occurs and measure the Reversion Packet Loss [TERM-ID].
13. Calculate the Reversion Time [TERM-ID] benchmark using the selected Failover Time Calculation Method (TBLM, PLBM, or TBM) [TERM-ID].
14. Verify Headend signals new LSP and protection should be in place again.

IT is RECOMMENDED that this procedure be repeated for each of the link failure triggers defined in section 5.1.

7.3. Mid-Point PLR with Link Failure

Objective:

To benchmark the MPLS failover time due to link failure events described in section 5.1 experienced by the DUT which is the Mid-Point PLR.

Test Setup:

- A. Select any one topology out of the 8 from section 6.
- B. Select overlay technology for FRR test as Mid-Point LSPs.
- C. The DUT will also have 2 interfaces connected to the traffic generator.

Test Configuration:

- 1. Configure the number of primaries on R1 and the backups on R2 as required by the topology selected.
- 2. Configure the test setup to support Reversion.
- 3. Advertise prefixes (as per FRR Scalability Table described in Appendix A) by the tail end.

Procedure:

Test Case "7.1.2. Mid-Point PLR Forwarding Performance" MUST be completed first to obtain the Throughput to use as the offered load.

- 1. Establish the primary LSP on R1 required by the topology selected.
- 2. Establish the backup LSP on R2 required by the selected topology.
- 3. Perform steps 3 through 14 from section 7.2 Headend PLR with Link Failure.

IT is RECOMMENDED that this procedure be repeated for each of the link failure triggers defined in section 5.1.

7.4. Headend PLR with Node Failure

Objective:

To benchmark the MPLS failover time due to Node failure events described in section 5.1 experienced by the DUT which is the Headend PLR.

Test Setup:

- A. Select any one topology from section 6.
- B. Select overlay technology for FRR test (e.g. IGP, VPN, or VC).
- C. The DUT will also have 2 interfaces connected to the traffic generator/analyzer.

Test Configuration:

- 1. Configure the number of primaries on R2 and the backups on R2 as required by the topology selected.
- 2. Configure the test setup to support Reversion.
- 3. Advertise prefixes (as per FRR Scalability Table described in Appendix A) by the tail end.

Procedure:

Test Case "7.1.1. Headend PLR Forwarding Performance" MUST be completed first to obtain the Throughput to use as the offered load.

- 1. Establish the primary LSP on R2 required by the topology selected.
- 2. Establish the backup LSP on R2 required by the selected topology.
- 3. Verify primary and backup LSPs are up and that primary is protected.
- 4. Verify Fast Reroute protection.
- 5. Setup traffic streams for the offered load as described in section 5.7.
- 6. Provide the offered load from the tester at the Throughput [Br91] level obtained from test case 7.1.1.

7. Verify traffic is switched over Primary LSP without packet loss.
8. Trigger a node failure as described in section 5.1.
9. Perform steps 9 through 14 in 7.2 Headend PLR with Link Failure.

IT is RECOMMENDED that this procedure be repeated for each of the node failure triggers defined in section 5.1.

7.5. Mid-Point PLR with Node Failure

Objective:

To benchmark the MPLS failover time due to Node failure events described in section 5.1 experienced by the DUT which is the Mid-Point PLR.

Test Setup:

- A. Select any one topology from section 6.1 to 6.2.
- B. Select overlay technology for FRR test as Mid-Point LSPs.
- C. The DUT will also have 2 interfaces connected to the traffic generator.

Test Configuration:

1. Configure the number of primaries on R1 and the backups on R2 as required by the topology selected.
2. Configure the test setup to support Reversion.
3. Advertise prefixes (as per FRR Scalability Table described in Appendix A) by the tail end.

Procedure:

Test Case "7.1.1. Mid-Point PLR Forwarding Performance" MUST be completed first to obtain the Throughput to use as the offered load.

1. Establish the primary LSP on R1 required by the topology selected.
2. Establish the backup LSP on R2 required by the selected topology.
3. Verify primary and backup LSPs are up and that primary is protected.
4. Verify Fast Reroute protection.
5. Setup traffic streams for the offered load as described in section 5.7.
6. Provide the offered load from the tester at the Throughput [Br91] level obtained from test case 7.1.1.
7. Verify traffic is switched over Primary LSP without packet loss.
8. Trigger a node failure as described in section 5.1.
9. Perform steps 9 through 14 in 7.2 Headend PLR with Link Failure.

IT is RECOMMENDED that this procedure be repeated for each of the node failure triggers defined in section 5.1.

8. Reporting Format

For each test, it is recommended that the results be reported in the following format.

Parameter	Units
IGP used for the test	ISIS-TE/ OSPF-TE
Interface types	Gige,POS,ATM,VLAN etc.
Packet Sizes offered to the DUT	Bytes (at layer 3)
Offered Load	packets per second

IGP routes advertised	Number of IGP routes
Penultimate Hop Popping	Used/Not Used
RSVP hello timers	Milliseconds
Number of Protected tunnels	Number of tunnels
Number of VPN routes installed on the Headend	Number of VPN routes
Number of VC tunnels	Number of VC tunnels
Number of mid-point tunnels	Number of tunnels
Number of Prefixes protected by Primary	Number of LSPs
Topology being used	Section number, and figure reference
Failover Event	Event type
Re-optimization	Yes/No

Benchmarks (to be recorded for each test case):

Failover-

Failover Time	seconds
Failover Packet Loss	packets
Additive Backup Delay	seconds
Out-of-Order Packets	packets
Duplicate Packets	packets
Failover Time Calculation Method	Method Used

Reversion-

Reversion Time	seconds
Reversion Packet Loss	packets
Additive Backup Delay	seconds
Out-of-Order Packets	packets
Duplicate Packets	packets
Failover Time Calculation Method	Method Used

Failover Time suggested above is calculated using one of the following three methods

1. Packet-Loss Based method (PLBM): (Number of packets dropped/ packets per second * 1000) milliseconds. This method could also be referred as Loss-Derived method.
2. Time-Based Loss Method (TBLM): This method relies on the ability of the Traffic generators to provide statistics which reveal the duration of failure in milliseconds based on when the packet loss occurred (interval between non-zero packet loss and zero loss).
3. Timestamp Based Method (TBM): This method of failover calculation is based on the timestamp that gets transmitted as payload in the packets originated by the generator. The Traffic Analyzer records the timestamp of the last packet received before the failover event and the first packet after the failover and derives the time based on the difference between these 2 timestamps. Note: The payload could also contain sequence numbers for out-of-order packet calculation and duplicate packets.

The timestamp based method method would be able to detect Reversion impairments beyond loss, thus it is RECOMMENDED method as a Failover Time method.

9. Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the constraints specified in the sections above.

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT/SUT.

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

10. IANA Considerations

This draft does not require any new allocations by IANA.

11. Acknowledgements

We would like to thank Jean Philip Vasseur for his invaluable input to the document, Curtis Villamizar for his contribution in suggesting text on definition and need for benchmarking Correlated failures and Bhavani Parise for his textual input and review. Additionally we would like to thank Al Morton, Arun Gandhi, Amrit Hanspal, Karu Ratnam, Raveesh Janardan, Andrey Kiselev, and Mohan Nanduri for their formal reviews of this document.

12. References

12.1. Informative References

- [IGP-METH] Poretsky, S., Imhoff, B., and K. Michielsen, "Terminology for Benchmarking Link-State IGP Data Plane Route Convergence", draft-ietf-bmwg-igp-dataplane-conv-term-23 (work in progress), February 2011.
- [Br91] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [Ma98] Mandeville, R., "Benchmarking Terminology for LAN Switching Devices", RFC 2285, February 1998.
- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [MPLS-FRR-EXT] Pan, P., Swallow, G., and A. Atlas, "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090, May 2005.
- [Po06] Poretsky, S., Perser, J., Erramilli, S., and S. Khurana, "Terminology for Benchmarking Network-layer Traffic Control Mechanisms", RFC 4689, October 2006.
- [MPLS-FWD] Akhter, A., Asati, R., and C. Pignataro, "MPLS Forwarding Benchmarking Methodology for IP Flows", RFC 5695, November 2009.
- [RFC6414] Papneja, R., Poretsky, S., Vapiwala, S., and J. Karthik, "Benchmarking Terminology for Protection Performance", RFC 6414, October 2011.

12.2. Normative References

- [Br97] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

Appendix A. Fast Reroute Scalability Table

This section provides the recommended numbers for evaluating the scalability of fast reroute implementations. It also recommends the typical numbers for IGP/VPNv4 Prefixes, LSP Tunnels and VC entries. Based on the features supported by the device under test (DUT), appropriate scaling limits can be used for the test bed.

A1. FRR IGP Table

No. of Headend TE Tunnels	IGP Prefixes
1	100
1	500
1	1000
1	2000
1	5000
2 (Load Balance)	100
2 (Load Balance)	500
2 (Load Balance)	1000
2 (Load Balance)	2000
2 (Load Balance)	5000
100	100
500	500
1000	1000
2000	2000

A2. FRR VPN Table

No. of Headend TE Tunnels	VPNv4 Prefixes
1	100
1	500
1	1000
1	2000
1	5000
1	10000
1	20000
1	Max
2 (Load Balance)	100
2 (Load Balance)	500
2 (Load Balance)	1000
2 (Load Balance)	2000
2 (Load Balance)	5000
2 (Load Balance)	10000
2 (Load Balance)	20000
2 (Load Balance)	Max

A3. FRR Mid-Point LSP Table

No of Mid-point TE LSPs could be configured at recommended levels - 100, 500, 1000, 2000, or max supported number.

A2. FRR VC Table

No. of Headend TE Tunnels	VC entries
1	100
1	500
1	1000
1	2000
1	Max
100	100
500	500
1000	1000
2000	2000

Appendix B. Abbreviations

BFD	- Bidirectional Fault Detection
BGP	- Border Gateway protocol
CE	- Customer Edge
DUT	- Device Under Test
FRR	- Fast Reroute
IGP	- Interior Gateway Protocol
IP	- Internet Protocol
LSP	- Label Switched Path
MP	- Merge Point
MPLS	- Multi Protocol Label Switching
N-Nhop	- Next - Next Hop
Nhop	- Next Hop
OIR	- Online Insertion and Removal
P	- Provider
PE	- Provider Edge
PHP	- Penultimate Hop Popping
PLR	- Point of Local Repair
RSVP	- Resource reSerVation Protocol
SRLG	- Shared Risk Link Group
TA	- Traffic Analyzer
TE	- Traffic Engineering
TG	- Traffic Generator
VC	- Virtual Circuit
VPN	- Virtual Private Network

Authors' Addresses

Rajiv Papneja
Huawei Technologies
2330 Central Expressway
Santa Clara, CA 95050
USA

Email: rajiv.papneja@huawei.com

Samir Vapiwala
Cisco Systems
300 Beaver Brook Road
Boxborough, MA 01719
USA

Email: svapiwal@cisco.com

Jay Karthik
Cisco Systems
300 Beaver Brook Road
Boxborough, MA 01719
USA

Email: jkarthik@cisco.com

Scott Poretsky
Allot Communications
USA

Email: sporetsky@allot.com

Shankar Rao
5005 E. Dartmouth Ave.
Denver, CO 80222
USA

Email: srao@du.edu

Jean-Louis Le Roux
France Telecom
2 av Pierre Marzin
22300 Lannion
France

Email: jeanlouis.leroux@francetelecom.com

Benchmarking Methodology Working Group
Internet-Draft
Intended status: Informational
Expires: May 16, 2015

C. Davids
Illinois Institute of Technology
V. Gurbani
Bell Laboratories,
Alcatel-Lucent
S. Poretsky
Allot Communications
November 12, 2014

Methodology for Benchmarking Session Initiation Protocol (SIP) Devices:
Basic session setup and registration
draft-ietf-bmwg-sip-bench-meth-12

Abstract

This document provides a methodology for benchmarking the Session Initiation Protocol (SIP) performance of devices. Terminology related to benchmarking SIP devices is described in the companion terminology document. Using these two documents, benchmarks can be obtained and compared for different types of devices such as SIP Proxy Servers, Registrars and Session Border Controllers. The term "performance" in this context means the capacity of the device-under-test (DUT) to process SIP messages. Media streams are used only to study how they impact the signaling behavior. The intent of the two documents is to provide a normalized set of tests that will enable an objective comparison of the capacity of SIP devices. Test setup parameters and a methodology are necessary because SIP allows a wide range of configuration and operational conditions that can influence performance benchmark measurements.

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 <http://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 May 16, 2015.

Copyright Notice

Copyright (c) 2014 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Terminology	4
2. Introduction	4
3. Benchmarking Topologies	5
4. Test Setup Parameters	7
4.1. Selection of SIP Transport Protocol	7
4.2. Connection-oriented Transport Management	7
4.3. Signaling Server	8
4.4. Associated Media	8
4.5. Selection of Associated Media Protocol	8
4.6. Number of Associated Media Streams per SIP Session	8
4.7. Codec Type	8
4.8. Session Duration	8
4.9. Attempted Sessions per Second (sps)	9
4.10. Benchmarking algorithm	9
5. Reporting Format	11
5.1. Test Setup Report	11
5.2. Device Benchmarks for session setup	12
5.3. Device Benchmarks for registrations	12
6. Test Cases	13
6.1. Baseline Session Establishment Rate of the test bed	13
6.2. Session Establishment Rate without media	13
6.3. Session Establishment Rate with Media not on DUT	13
6.4. Session Establishment Rate with Media on DUT	14
6.5. Session Establishment Rate with TLS Encrypted SIP	14
6.6. Session Establishment Rate with IPsec Encrypted SIP	15
6.7. Registration Rate	15
6.8. Re-Registration Rate	16
7. IANA Considerations	16
8. Security Considerations	16
9. Acknowledgments	17
10. References	17
10.1. Normative References	17
10.2. Informative References	17
Appendix A. R Code Component to simulate benchmarking algorithm	18
Authors' Addresses	20

1. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in BCP 14, conforming to [RFC2119] and indicate requirement levels for compliant implementations.

RFC 2119 defines the use of these key words to help make the intent of standards track documents as clear as possible. While this document uses these keywords, this document is not a standards track document. The term Throughput is defined in [RFC2544].

Terms specific to SIP [RFC3261] performance benchmarking are defined in [I-D.sip-bench-term].

2. Introduction

This document describes the methodology for benchmarking Session Initiation Protocol (SIP) performance as described in the Terminology document [I-D.sip-bench-term]. The methodology and terminology are to be used for benchmarking signaling plane performance with varying signaling and media load. Media streams, when used, are used only to study how they impact the signaling behavior. This document concentrates on benchmarking SIP session setup and SIP registrations only.

The device-under-test (DUT) is a RFC3261-capable [RFC3261] network intermediary that plays the role of a registrar, redirect server, stateful proxy, a Session Border Controller (SBC) or a B2BUA. This document does not require the intermediary to assume the role of a stateless proxy. Benchmarks can be obtained and compared for different types of devices such as a SIP proxy server, Session Border Controllers (SBC), SIP registrars and a SIP proxy server paired with a media relay.

The test cases provide metrics for benchmarking the maximum 'SIP Registration Rate' and maximum 'SIP Session Establishment Rate' that the DUT can sustain over an extended period of time without failures (extended period of time is defined in the algorithm in Section 4.10). Some cases are included to cover encrypted SIP. The test topologies that can be used are described in the Test Setup section. Topologies in which the DUT handles media as well as those in which the DUT does not handle media are both considered. The measurement of the performance characteristics of the media itself is outside the scope of these documents.

Benchmark metrics could possibly be impacted by Associated Media. The selected values for Session Duration and Media Streams per Session enable benchmark metrics to be benchmarked without Associated Media. Session Setup Rate could possibly be impacted by the selected value for Maximum Sessions Attempted. The benchmark for Session Establishment Rate is measured with a fixed value for maximum Session Attempts.

Finally, the overall value of these tests is to serve as a comparison function between multiple SIP implementations. One way to use these tests is to derive benchmarks with SIP devices from Vendor-A, derive a new set of benchmarks with similar SIP devices from Vendor-B and perform a comparison on the results of Vendor-A and Vendor-B. This document does not make any claims on the interpretation of such results.

3. Benchmarking Topologies

Test organizations need to be aware that these tests generate large volumes of data and consequently ensure that networking devices like hubs, switches or routers are able to handle the generated volume.

The test cases enumerated in Section 6.1 to Section 6.6 operate on two test topologies: one in which the DUT does not process the media (Figure 1) and the other in which it does process media (Figure 2). In both cases, the tester or emulated agent (EA) sends traffic into the DUT and absorbs traffic from the DUT. The diagrams in Figure 1 and Figure 2 represent the logical flow of information and do not dictate a particular physical arrangement of the entities.

Figure 1 depicts a layout in which the DUT is an intermediary between the two interfaces of the EA. If the test case requires the exchange of media, the media does not flow through the DUT but rather passes directly between the two endpoints. Figure 2 shows the DUT as an intermediary between the two interfaces of the EA. If the test case requires the exchange of media, the media flows through the DUT between the endpoints.

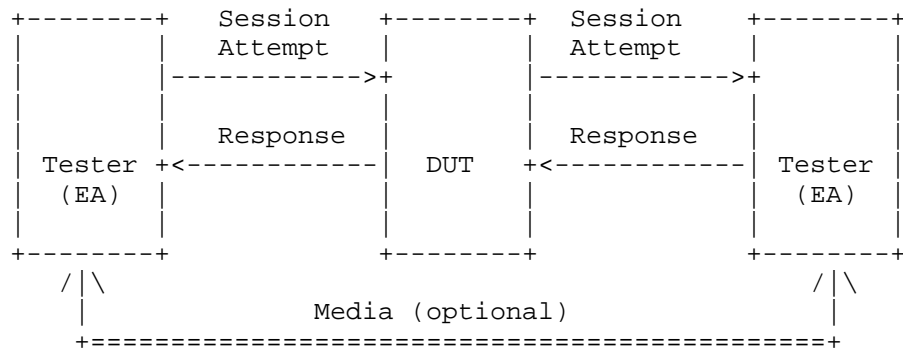


Figure 1: DUT as an intermediary, end-to-end media

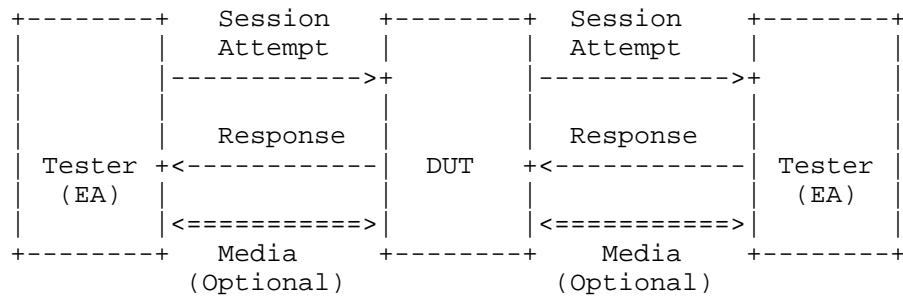


Figure 2: DUT as an intermediary forwarding media

The test cases enumerated in Section 6.7 and Section 6.8 use the topology in Figure 3 below.

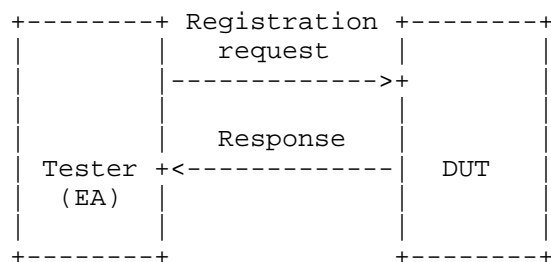


Figure 3: Registration and Re-registration tests

During registration or re-registration, the DUT may involve backend network elements and data stores. These network elements and data stores are not shown in Figure 3, but it is understood that they will impact the time required for the DUT to generate a response.

This document explicitly separates a registration test (Section 6.7) from a re-registration test (Section 6.8) because in certain networks, the time to re-register may vary from the time to perform an initial registration due to the backend processing involved. It is expected that the registration tests and the re-registration test will be performed with the same set of backend network elements in order to derive a stable metric.

4. Test Setup Parameters

4.1. Selection of SIP Transport Protocol

Test cases may be performed with any transport protocol supported by SIP. This includes, but is not limited to, TCP, UDP, TLS and websockets. The protocol used for the SIP transport protocol must be reported with benchmarking results.

SIP allows a DUT to use different transports for signaling on either side of the connection to the EAs. Therefore, this document assumes that the same transport is used on both sides of the connection; if this is not the case in any of the tests, the transport on each side of the connection **MUST** be reported in the test reporting template.

4.2. Connection-oriented Transport Management

SIP allows a device to open one connection and send multiple requests over the same connection (responses are normally received over the same connection that the request was sent out on). The protocol also allows a device to open a new connection for each individual request. A connection management strategy will have an impact on the results obtained from the test cases, especially for connection-oriented transports such as TLS. For such transports, the cryptographic handshake must occur every time a connection is opened.

The connection management strategy, i.e., use of one connection to send all requests or closing an existing connection and opening a new connection to send each request, **MUST** be reported with the benchmarking result.

4.3. Signaling Server

The Signaling Server is defined in the companion terminology document, ([I-D.sip-bench-term], Section 3.2.2). The Signaling Server is a DUT.

4.4. Associated Media

Some tests require Associated Media to be present for each SIP session. The test topologies to be used when benchmarking DUT performance for Associated Media are shown in Figure 1 and Figure 2.

4.5. Selection of Associated Media Protocol

The test cases specified in this document provide SIP performance independent of the protocol used for the media stream. Any media protocol supported by SIP may be used. This includes, but is not limited to, RTP, and SRTP. The protocol used for Associated Media MUST be reported with benchmarking results.

4.6. Number of Associated Media Streams per SIP Session

Benchmarking results may vary with the number of media streams per SIP session. When benchmarking a DUT for voice, a single media stream is used. When benchmarking a DUT for voice and video, two media streams are used. The number of Associated Media Streams MUST be reported with benchmarking results.

4.7. Codec Type

The test cases specified in this document provide SIP performance independent of the media stream codec. Any codec supported by the EAs may be used. The codec used for Associated Media MUST be reported with the benchmarking results.

4.8. Session Duration

The value of the DUT's performance benchmarks may vary with the duration of SIP sessions. Session Duration MUST be reported with benchmarking results. A Session Duration of zero seconds indicates transmission of a BYE immediately following a successful SIP establishment. Setting this parameter to the value '0' indicates that a BYE will be sent by the EA immediately after the EA receives a 200 OK to the INVITE. Setting this parameter to a time value greater than the duration of the test indicates that a BYE is never sent.

4.9. Attempted Sessions per Second (sps)

The value of the DUT's performance benchmarks may vary with the Session Attempt Rate offered by the tester. Session Attempt Rate MUST be reported with the benchmarking results.

The test cases enumerated in Section 6.1 to Section 6.6 require that the EA is configured to send the final 2xx-class response as quickly as it can. This document does not require the tester to add any delay between receiving a request and generating a final response.

4.10. Benchmarking algorithm

In order to benchmark the test cases uniformly in Section 6, the algorithm described in this section should be used. A prosaic description of the algorithm and a pseudo-code description are provided below, and a simulation written in the R statistical language [Rtool] is provided in Appendix A.

The goal is to find the largest value, *R*, a SIP Session Attempt Rate, measured in sessions-per-second (sps), which the DUT can process with zero errors over a defined, extended period. This period is defined as the amount of time needed to attempt *N* SIP sessions, where *N* is a parameter of test, at the attempt rate, *R*. An iterative process is used to find this rate. The algorithm corresponding to this process converges to *R*.

If the DUT vendor provides a value for *R*, the tester can use this value. In cases where the DUT vendor does not provide a value for *R*, or where the tester wants to establish the *R* of a system using local media characteristics, the algorithm should be run by setting "*r*", the session attempt rate, equal to a value of the tester's choice. For example the tester may initialize "*r* = 100" to start the algorithm and observe the value at convergence. The algorithm dynamically increases and decreases "*r*" as it converges to the a maximum sps value for *R*. The dynamic increase and decrease rate is controlled by the weights "*w*" and "*d*", respectively.

The pseudo-code corresponding to the description above follows, and a simulation written in the R statistical language is provided in Appendix A.

```
; ---- Parameters of test, adjust as needed
N := 50000 ; Global maximum; once largest session rate has
           ; been established, send this many requests before
           ; calling the test a success
m := {...} ; Other attributes that affect testing, such
```

```

; as media streams, etc.
r  := 100    ; Initial session attempt rate (in sessions/sec).
; Adjust as needed (for example, if DUT can handle
; thousands of calls in steady state, set to
; appropriate value in the thousands).
w  := 0.10   ; Traffic increase weight (0 < w <= 1.0)
d  := max(0.10, w / 2) ; Traffic decrease weight

; ---- End of parameters of test

proc find_R

    R = max_sps(r, m, N) ; Setup r sps, each with m media
; characteristics until N sessions have been attempted.
; Note that if a DUT vendor provides this number, the tester
; can use the number as a Session Attempt Rate, R, instead
; of invoking max_sps()

end proc

; Iterative process to figure out the largest number of
; sps that we can achieve in order to setup n sessions.
; This function converges to R, the Session Attempt Rate.
proc max_sps(r, m, n)
    s      := 0    ; session setup rate
    old_r  := 0    ; old session setup rate
    h      := 0    ; Return value, R
    count  := 0

    ; Note that if w is small (say, 0.10) and r is small
    ; (say, <= 9), the algorithm will not converge since it
    ; uses floor() to increment r dynamically. It is best
    ; off to start with the defaults (w = 0.10 and
    ; r >= 100)

    while (TRUE) {
        s := send_traffic(r, m, n) ; Send r sps, with m media
; characteristics until n sessions have been attempted.
        if (s == n) {
            if (r > old_r) {
                old_r = r
            }
        }
        else {
            count = count + 1
            if (count >= 10) {
                # We've converged.
                h := max(r, old_r)
                break
            }
        }
    }
end proc
```

```

        }
    }
    r := floor(r + (w * r))
}
else {
    r := floor(r - (d * r))
    d := max(0.10, d / 2)
    w := max(0.10, w / 2)
}
}
return h
end proc

```

5. Reporting Format

5.1. Test Setup Report

SIP Transport Protocol = _____
 (valid values: TCP|UDP|TLS|SCTP|websockets|specify-other)
 (specify if same transport used for connections to the DUT
 and connections from the DUT. If different transports
 used on each connection, enumerate the transports used)

Connection management strategy for connection oriented
 transports

DUT receives requests on one connection = _____
 (Yes or no. If no, DUT accepts a new connection for
 every incoming request, sends a response on that
 connection and closes the connection)
 DUT sends requests on one connection = _____
 (yes or no. If no, DUT initiates a new connection to
 send out each request, gets a response on that
 connection and closes the connection)

Session Attempt Rate _____
 (Session attempts/sec)
 (The initial value for "r" in Benchmarking Algorithm of
 Section 4.10)

Session Duration = _____
 (In seconds)

Total Sessions Attempted = _____
(Total sessions to be created over duration of test)

Media Streams Per Session = _____
(number of streams per session)

Associated Media Protocol = _____
(RTP|SRTP|specify-other)

Codec = _____
(Codec type as identified by the organization that specifies the codec)

Media Packet Size (audio only) = _____
(Number of bytes in an audio packet)

Establishment Threshold time = _____
(Seconds)

TLS ciphersuite used
(for tests involving TLS) = _____
(E.g., TLS_RSA_WITH_AES_128_CBC_SHA)

IPSec profile used
(For tests involving IPSEC) = _____

5.2. Device Benchmarks for session setup

Session Establishment Rate, "R" = _____
(sessions per second)
Is DUT acting as a media relay (yes/no) = _____

5.3. Device Benchmarks for registrations

Registration Rate = _____
(registrations per second)

Re-registration Rate = _____
(registrations per second)

Notes = _____
(List any specific backend processing required or other parameters that may impact the rate)

6. Test Cases

6.1. Baseline Session Establishment Rate of the test bed

Objective:

To benchmark the Session Establishment Rate of the Emulated Agent (EA) with zero failures.

Procedure:

1. Configure the DUT in the test topology shown in Figure 1.
2. Set media streams per session to 0.
3. Execute benchmarking algorithm as defined in Section 4.10 to get the baseline session establishment rate. This rate MUST be recorded using any pertinent parameters as shown in the reporting format of Section 5.1.

Expected Results: This is the scenario to obtain the maximum Session Establishment Rate of the EA and the test bed when no DUT is present. The results of this test might be used to normalize test results performed on different test beds or simply to better understand the impact of the DUT on the test bed in question.

6.2. Session Establishment Rate without media

Objective:

To benchmark the Session Establishment Rate of the DUT with no associated media and zero failures.

Procedure:

1. Configure a DUT according to the test topology shown in Figure 1 or Figure 2.
2. Set media streams per session to 0.
3. Execute benchmarking algorithm as defined in Section 4.10 to get the session establishment rate. This rate MUST be recorded using any pertinent parameters as shown in the reporting format of Section 5.1.

Expected Results: Find the Session Establishment Rate of the DUT when the EA is not sending media streams.

6.3. Session Establishment Rate with Media not on DUT

Objective:

To benchmark the Session Establishment Rate of the DUT with zero failures when Associated Media is included in the benchmark test but the media is not running through the DUT.

Procedure:

1. Configure a DUT according to the test topology shown in Figure 1.
2. Set media streams per session to 1.
3. Execute benchmarking algorithm as defined in Section 4.10 to get the session establishment rate with media. This rate **MUST** be recorded using any pertinent parameters as shown in the reporting format of Section 5.1.

Expected Results: Session Establishment Rate results obtained with Associated Media with any number of media streams per SIP session are expected to be identical to the Session Establishment Rate results obtained without media in the case where the DUT is running on a platform separate from the Media Relay.

6.4. Session Establishment Rate with Media on DUT

Objective:

To benchmark the Session Establishment Rate of the DUT with zero failures when Associated Media is included in the benchmark test and the media is running through the DUT.

Procedure:

1. Configure a DUT according to the test topology shown in Figure 2.
2. Set media streams per session to 1.
3. Execute benchmarking algorithm as defined in Section 4.10 to get the session establishment rate with media. This rate **MUST** be recorded using any pertinent parameters as shown in the reporting format of Section 5.1.

Expected Results: Session Establishment Rate results obtained with Associated Media may be lower than those obtained without media in the case where the DUT and the Media Relay are running on the same platform. It may be helpful for the tester to be aware of the reasons for this degradation, although these reasons are not parameters of the test. For example, the degree of performance degradation may be due to what the DUT does with the media (e.g., relaying vs. transcoding), the type of media (audio vs. video vs. data), and the codec used for the media. There may also be cases where there is no performance impact, if the DUT has dedicated media-path hardware.

6.5. Session Establishment Rate with TLS Encrypted SIP

Objective:

To benchmark the Session Establishment Rate of the DUT with zero failures when using TLS encrypted SIP signaling.

Procedure:

1. If the DUT is being benchmarked as a proxy or B2BUA, then configure the DUT in the test topology shown in Figure 1 or Figure 2.
2. Configure the tester to enable TLS over the transport being used during benchmarking. Note the ciphersuite being used for TLS and record it in Section 5.1.
3. Set media streams per session to 0 (media is not used in this test).
4. Execute benchmarking algorithm as defined in Section 4.10 to get the session establishment rate with TLS encryption.

Expected Results: Session Establishment Rate results obtained with TLS Encrypted SIP may be lower than those obtained with plaintext SIP.

6.6. Session Establishment Rate with IPsec Encrypted SIP**Objective:**

To benchmark the Session Establishment Rate of the DUT with zero failures when using IPsec Encrypted SIP signaling.

Procedure:

1. Configure a DUT according to the test topology shown in Figure 1 or Figure 2.
2. Set media streams per session to 0 (media is not used in this test).
3. Configure tester for IPSec. Note the IPSec profile being used for and record it in Section 5.1.
4. Execute benchmarking algorithm as defined in Section 4.10 to get the session establishment rate with encryption.

Expected Results: Session Establishment Rate results obtained with IPSec Encrypted SIP may be lower than those obtained with plaintext SIP.

6.7. Registration Rate**Objective:**

To benchmark the maximum registration rate the DUT can handle over an extended time period with zero failures.

Procedure:

1. Configure a DUT according to the test topology shown in Figure 3.
2. Set the registration timeout value to at least 3600 seconds.
3. Each register request MUST be made to a distinct address of record (AoR). Execute benchmarking algorithm as defined in Section 4.10 to get the maximum registration rate. This rate MUST be recorded using any pertinent parameters as shown in the reporting format of Section 5.1. For example, the use of TLS or IPSec during registration must be noted in the reporting format. In the same vein, any specific backend processing (use of databases, authentication servers, etc.) SHOULD be recorded as well.

Expected Results: Provides a maximum registration rate.

6.8. Re-Registration Rate

Objective:

To benchmark the re-registration rate of the DUT with zero failures using the same backend processing and parameters used during Section 6.7.

Procedure:

1. Configure a DUT according to the test topology shown in Figure 3.
2. First, execute test detailed in Section 6.7 to register the endpoints with the registrar and obtain the registration rate.
3. After at least 5 minutes of Step 2, but no more than 10 minutes after Step 2 has been performed, re-register the same AoRs used in Step 3 of Section 6.7. This will count as a re-registration because the SIP AoRs have not yet expired.

Expected Results: Note the rate obtained through this test for comparison with the rate obtained in Section 6.7.

7. IANA Considerations

This document does not requires any IANA considerations.

8. Security Considerations

Documents of this type do not directly affect the security of Internet or corporate networks as long as benchmarking is not performed on devices or systems connected to production networks.

Security threats and how to counter these in SIP and the media layer is discussed in RFC3261, RFC3550, and RFC3711 and various other drafts. This document attempts to formalize a set of common methodology for benchmarking performance of SIP devices in a lab environment.

9. Acknowledgments

The authors would like to thank Keith Drage and Daryl Malas for their contributions to this document. Dale Worley provided an extensive review that lead to improvements in the documents. We are grateful to Barry Constantine, William Cerveney and Robert Sparks for providing valuable comments during the document's last calls and expert reviews. Al Morton and Sarah Banks have been exemplary working group chairs, we thank them for tracking this work to completion. Tom Taylor provided an in-depth review and subsequent comments on the benchmarking convergence algorithm in Section 4.10.

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [I-D.sip-bench-term]
Davids, C., Gurbani, V., and S. Poretsky, "SIP Performance Benchmarking Terminology",
draft-ietf-bmwg-sip-bench-term-12 (work in progress),
November 2014.

10.2. Informative References

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.
- [Rtool] R Development Core Team, "R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org/>, , 2011.

Appendix A. R Code Component to simulate benchmarking algorithm

```
# Copyright (c) 2014 IETF Trust and Vijay K. Gurbani. All
# rights reserved.
#
# Redistribution and use in source and binary forms, with
# or without modification, are permitted provided that the
# following conditions are met:
#
# * Redistributions of source code must retain the above
#   copyright notice, this list of conditions and the following
#   disclaimer.
# * Redistributions in binary form must reproduce the above
#   copyright notice, this list of conditions and the following
#   disclaimer in the documentation and/or other materials
#   provided with the distribution.
# * Neither the name of Internet Society, IETF or IETF Trust,
#   nor the names of specific contributors, may be used
#   to endorse or promote products derived from this software
#   without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
# DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
# CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
# SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
# BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
# INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
# WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
# NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE
# USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
# OF SUCH DAMAGE.

w = 0.10
d = max(0.10, w / 2)
DUT_max_sps = 460      # Change as needed to set the max sps value
                       # for a DUT

# Returns R, given r (initial session attempt rate).
# E.g., assume that a DUT handles 460 sps in steady state
# and you have saved this code in a file simulate.r. Then,
# start an R session and do the following:
#
# > source("simulate.r")
```

```

# > find_R(100)
# ... debug output omitted ...
# [1] 458
#
# Thus, the max sps that the DUT can handle is 458 sps, which is
# close to the absolute maximum of 460 sps the DUT is specified to
# do.
find_R <- function(r) {
  s      = 0
  old_r  = 0
  h      = 0
  count  = 0

  # Note that if w is small (say, 0.10) and r is small
  # (say, <= 9), the algorithm will not converge since it
  # uses floor() to increment r dynamically. It is best
  # off to start with the defaults (w = 0.10 and
  # r >= 100)

  cat("r    old_r    w      d \n")
  while (TRUE) {
    cat(r, ' ', old_r, ' ', w, ' ', d, '\n')
    s = send_traffic(r)
    if (s == TRUE) {      # All sessions succeeded

      if (r > old_r) {
        old_r = r
      }
      else {
        count = count + 1

        if (count >= 10) {
          # We've converged.
          h = max(r, old_r)
          break
        }
      }

      r = floor(r + (w * r))
    }
    else {
      r = floor(r - (d * r))
      d = max(0.10, d / 2)
      w = max(0.10, w / 2)
    }
  }

  h
}

```



```
    }  
  
    send_traffic <- function(r) {  
      n = TRUE  
  
      if (r > DUT_max_sps) {  
        n = FALSE  
      }  
  
      n  
    }  
  }
```

Authors' Addresses

Carol Davids
Illinois Institute of Technology
201 East Loop Road
Wheaton, IL 60187
USA

Phone: +1 630 682 6024
Email: davids@iit.edu

Vijay K. Gurbani
Bell Laboratories, Alcatel-Lucent
1960 Lucent Lane
Rm 9C-533
Naperville, IL 60566
USA

Phone: +1 630 224 0216
Email: vkg@bell-labs.com

Scott Poretsky
Allot Communications
300 TradeCenter, Suite 4680
Woburn, MA 08101
USA

Phone: +1 508 309 2179
Email: sporetsky@allot.com

Benchmarking Methodology Working Group
Internet-Draft
Intended status: Informational
Expires: May 16, 2015

C. Davids
Illinois Institute of Technology
V. Gurbani
Bell Laboratories,
Alcatel-Lucent
S. Poretsky
Allot Communications
November 12, 2014

Terminology for Benchmarking Session Initiation Protocol (SIP) Devices:
Basic session setup and registration
draft-ietf-bmwg-sip-bench-term-12

Abstract

This document provides a terminology for benchmarking the Session Initiation Protocol (SIP) performance of devices. Methodology related to benchmarking SIP devices is described in the companion methodology document. Using these two documents, benchmarks can be obtained and compared for different types of devices such as SIP Proxy Servers, Registrars and Session Border Controllers. The term "performance" in this context means the capacity of the device-under-test (DUT) to process SIP messages. Media streams are used only to study how they impact the signaling behavior. The intent of the two documents is to provide a normalized set of tests that will enable an objective comparison of the capacity of SIP devices. Test setup parameters and a methodology is necessary because SIP allows a wide range of configuration and operational conditions that can influence performance benchmark measurements. A standard terminology and methodology will ensure that benchmarks have consistent definition and were obtained following the same procedures.

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 <http://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 May 16, 2015.

Copyright Notice

Copyright (c) 2014 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Terminology	4
2. Introduction	4
2.1. Scope	6
3. Term Definitions	7
3.1. Protocol Components	7
3.1.1. Session	7
3.1.2. Signaling Plane	8
3.1.3. Media Plane	8
3.1.4. Associated Media	9
3.1.5. Overload	9
3.1.6. Session Attempt	10
3.1.7. Established Session	10
3.1.8. Session Attempt Failure	11
3.2. Test Components	11
3.2.1. Emulated Agent	11
3.2.2. Signaling Server	12
3.2.3. SIP Transport Protocol	12
3.3. Test Setup Parameters	13
3.3.1. Session Attempt Rate	13
3.3.2. Establishment Threshold Time	13
3.3.3. Session Duration	14
3.3.4. Media Packet Size	14
3.3.5. Codec Type	15
3.4. Benchmarks	15
3.4.1. Session Establishment Rate	16
3.4.2. Registration Rate	16
3.4.3. Registration Attempt Rate	17
4. IANA Considerations	17
5. Security Considerations	17
6. Acknowledgments	18
7. References	18
7.1. Normative References	18
7.2. Informational References	19
Authors' Addresses	19

1. Terminology

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 BCP 14, RFC2119 [RFC2119]. RFC 2119 defines the use of these key words to help make the intent of standards track documents as clear as possible. While this document uses these keywords, this document is not a standards track document. The term Throughput is defined in RFC2544 [RFC2544].

For the sake of clarity and continuity, this document adopts the template for definitions set out in Section 2 of RFC 1242 [RFC1242].

The term Device Under Test (DUT) is defined in the following BMWG documents:

Device Under Test (DUT) (c.f., Section 3.1.1 RFC 2285 [RFC2285]).

Many commonly used SIP terms in this document are defined in RFC 3261 [RFC3261]. For convenience the most important of these are reproduced below. Use of these terms in this document is consistent with their corresponding definition in the base SIP specification [RFC3261] as amended by [RFC4320], [RFC5393] and [RFC6026].

- o Call Stateful: A proxy is call stateful if it retains state for a dialog from the initiating INVITE to the terminating BYE request. A call stateful proxy is always transaction stateful, but the converse is not necessarily true.
- o Stateful Proxy: A logical entity, as defined by [RFC3261], that maintains the client and server transaction state machines during the processing of a request. (Also known as a transaction stateful proxy.) The behavior of a stateful proxy is further defined in Section 16 of RFC 3261 [RFC3261]. A transaction stateful proxy is not the same as a call stateful proxy.
- o Back-to-back User Agent: A back-to-back user agent (B2BUA) is a logical entity that receives a request and processes it as a user agent server (UAS). In order to determine how the request should be answered, it acts as a user agent client (UAC) and generates requests. Unlike a proxy server, it maintains dialog state and must participate in all requests sent on the dialogues it has established. Since it is a concatenation of a UAC and a UAS, no explicit definitions are needed for its behavior.

2. Introduction

Service Providers and IT Organizations deliver Voice Over IP (VoIP) and Multimedia network services based on the IETF Session Initiation

Protocol (SIP) [RFC3261]. SIP is a signaling protocol originally intended to be used to dynamically establish, disconnect and modify streams of media between end users. As it has evolved it has been adopted for use in a growing number of services and applications. Many of these result in the creation of a media session, but some do not. Examples of this latter group include text messaging and subscription services. The set of benchmarking terms provided in this document is intended for use with any SIP-enabled device performing SIP functions in the interior of the network, whether or not these result in the creation of media sessions. The performance of end-user devices is outside the scope of this document.

A number of networking devices have been developed to support SIP-based VoIP services. These include SIP Servers, Session Border Controllers (SBC) and Back-to-back User Agents (B2BUA). These devices contain a mix of voice and IP functions whose performance may be reported using metrics defined by the equipment manufacturer or vendor. The Service Provider or IT Organization seeking to compare the performance of such devices will not be able to do so using these vendor-specific metrics, whose conditions of test and algorithms for collection are often unspecified.

SIP functional elements and the devices that include them can be configured many different ways and can be organized into various topologies. These configuration and topological choices impact the value of any chosen signaling benchmark. Unless these conditions-of-test are defined, a true comparison of performance metrics across multiple vendor implementations will not be possible.

Some SIP-enabled devices terminate or relay media as well as signaling. The processing of media by the device impacts the signaling performance. As a result, the conditions-of-test must include information as to whether or not the device under test processes media. If the device processes media during the test, a description of the media must be provided. This document and its companion methodology document [I-D.ietf-bmwg-sip-bench-meth] provide a set of black-box benchmarks for describing and comparing the performance of devices that incorporate the SIP User Agent Client and Server functions and that operate in the network's core.

The definition of SIP performance benchmarks necessarily includes definitions of Test Setup Parameters and a test methodology. These enable the Tester to perform benchmarking tests on different devices and to achieve comparable results. This document provides a common set of definitions for Test Components, Test Setup Parameters, and Benchmarks. All the benchmarks defined are black-box measurements of the SIP signaling plane. The Test Setup Parameters and Benchmarks defined in this document are intended for use with the companion

Methodology document.

2.1. Scope

The scope of this document is summarized as follows:

- o This terminology document describes SIP signaling performance benchmarks for black-box measurements of SIP networking devices. Stress and debug scenarios are not addressed in this document.
- o The DUT must be RFC 3261 capable network equipment. This may be a Registrar, Redirect Server, or Stateful Proxy. This document does not require the intermediary to assume the role of a stateless proxy. A DUT may also include a B2BUA, SBC functionality.
- o The Tester acts as multiple "Emulated Agents" (EA) that initiate (or respond to) SIP messages as session endpoints and source (or receive) associated media for established connections.
- o SIP Signaling in presence of media
 - * The media performance is not benchmarked.
 - * Some tests require media, but the use of media is limited to observing the performance of SIP signaling. Tests that require media will annotate the media characteristics as a condition of test.
 - * The type of DUT dictates whether the associated media streams traverse the DUT. Both scenarios are within the scope of this document.
 - * SIP is frequently used to create media streams; the signaling plane and media plane are treated as orthogonal to each other in this document. While many devices support the creation of media streams, benchmarks that measure the performance of these streams are outside the scope of this document and its companion methodology document [I-D.ietf-bmwg-sip-bench-meth]. Tests may be performed with or without the creation of media streams. The presence or absence of media streams MUST be noted as a condition of the test as the performance of SIP devices may vary accordingly. Even if the media is used during benchmarking, only the SIP performance will be benchmarked, not the media performance or quality.
- o Both INVITE and non-INVITE scenarios (registrations) are addressed in this document. However, benchmarking SIP presence or subscribe-notify extensions is not a part of this document.
- o Different transport -- such as UDP, TCP, SCTP, or TLS -- may be used. The specific transport mechanism MUST be noted as a condition of the test as the performance of SIP devices may vary accordingly.
- o REGISTER and INVITE requests may be challenged or remain unchallenged for authentication purpose. Whether or not the REGISTER and INVITE requests are challenged is a condition of test which will be recorded along with other such parameters which may impact the SIP performance of the device or system under test.

- o Re-INVITE requests are not considered in scope of this document since the benchmarks for INVITEs are based on the dialog created by the INVITE and not on the transactions that take place within that dialog.
- o Only session establishment is considered for the performance benchmarks. Session disconnect is not considered in the scope of this document. This is because our goal is to determine the maximum capacity of the device or system under test, that is the number of simultaneous SIP sessions that the device or system can support. It is true that there are BYE requests being created during the test process. These transactions do contribute to the load on the device or system under test and thus are accounted for in the metric we derive. We do not seek a separate metric for the number of BYE transactions a device or system can support.
- o IMS-specific scenarios are not considered, but test cases can be applied with 3GPP-specific SIP signaling and the P-CSCF as a DUT.
- o The benchmarks described in this document are intended for a laboratory environment and are not intended to be used on a production network. Some of the benchmarks send enough traffic that a denial of service attack is possible if used in production networks.

3. Term Definitions

3.1. Protocol Components

3.1.1. Session

Definition:

The combination of signaling and media messages and associated processing that enable a single SIP-based audio or video call, or SIP registration.

Discussion:

The term "session" commonly implies a media session. In this document the term is extended to cover the signaling and any media specified and invoked by the corresponding signaling.

Measurement Units:

N/A.

Issues:

None.

See Also:

- Media Plane
- Signaling Plane
- Associated Media

3.1.2. Signaling Plane

Definition:

The plane in which SIP messages [RFC3261] are exchanged between SIP Agents [RFC3261].

Discussion:

SIP messages are used to establish sessions in several ways: directly between two User Agents [RFC3261], through a Proxy Server [RFC3261], or through a series of Proxy Servers. The Session Description Protocol (SDP) is included in the Signaling Plane.

Measurement Units:

N/A.

Issues:

None.

See Also:

- Media Plane
- EAs

3.1.3. Media Plane

Definition:

The data plane in which one or more media streams and their associated media control protocols (e.g., RTCP [RFC3550]) are exchanged between User Agents after a media connection has been created by the exchange of signaling messages in the Signaling Plane.

Discussion:

Media may also be known as the "bearer channel". The Media Plane MUST include the media control protocol, if one is used, and the media stream(s). Examples of media are audio and video. The media streams are described in the SDP of the Signaling Plane.

Measurement Units:

N/A.

Issues:

None.

See Also:

Signaling Plane

3.1.4. Associated Media

Definition:

Media that corresponds to an 'm' line in the SDP payload of the Signaling Plane.

Discussion:

The format of the media is determined by the SDP attributes for the corresponding 'm' line.

Measurement Units:

N/A.

Issues:

None.

3.1.5. Overload

Definition:

Overload is defined as the state where a SIP server does not have sufficient resources to process all incoming SIP messages [RFC6357].

Discussion:

The distinction between an overload condition and other failure scenarios is outside the scope of black box testing and of this document. Under overload conditions, all or a percentage of Session Attempts will fail due to lack of resources. In black box testing the cause of the failure is not explored. The fact that a failure occurred for whatever reason, will trigger the tester to reduce the offered load, as described in the companion methodology document, [I-D.ietf-bmwg-sip-bench-meth]. SIP server resources may include CPU processing capacity, network bandwidth, input/output queues, or disk resources. Any combination of resources may be fully utilized when a SIP server (the DUT) is in the overload condition. For proxy-only (or intermediary) devices, it is expected that the proxy will be driven into overload based on the delivery rate of signaling requests.

Measurement Units:

N/A.

3.1.6. Session Attempt

Definition:

A SIP INVITE or REGISTER request sent by the EA that has not received a final response.

Discussion:

The attempted session may be either an invitation to an audio/video communication or a registration attempt. When counting the number of session attempts we include all requests that are rejected for lack of authentication information. The EA needs to record the total number of session attempts including those attempts that are routinely rejected by a proxy that requires the UA to authenticate itself. The EA is provisioned to deliver a specific number of session attempts per second. But the EA must also count the actual number of session attempts per given time interval.

Measurement Units:

N/A.

Issues:

None.

See Also:

Session

Session Attempt Rate

3.1.7. Established Session

Definition:

A SIP session for which the EA acting as the UE/UA has received a 200 OK message.

Discussion:

An Established Session may be either an invitation to an audio/video communication or a registration attempt. Early dialogues for INVITE requests are out of scope for this work.

Measurement Units:

N/A.

Issues:

None.

See Also:

None.

3.1.8. Session Attempt Failure

Definition:

A session attempt that does not result in an Established Session.

Discussion:

The session attempt failure may be indicated by the following observations at the EA:

1. Receipt of a SIP 3xx-, 4xx-, 5xx-, or 6xx-class response to a Session Attempt.
2. The lack of any received SIP response to a Session Attempt within the Establishment Threshold Time (c.f. Section 3.3.2).

Measurement Units:

N/A.

Issues:

None.

See Also:

Session Attempt

3.2. Test Components

3.2.1. Emulated Agent

Definition:

A device in the test topology that initiates/responds to SIP messages as one or more session endpoints and, wherever applicable, sources/receives Associated Media for Established Sessions.

Discussion:

The EA functions in the Signaling and Media Planes. The Tester may act as multiple EAs.

Measurement Units:

N/A

Issues:

None.

See Also:

Media Plane
Signaling Plane
Established Session
Associated Media

3.2.2. Signaling Server

Definition:

Device in the test topology that facilitates the creation of sessions between EAs. This device is the DUT.

Discussion:

The DUT is a RFC3261-capable network intermediary such as a Registrar, Redirect Server, Stateful Proxy, B2BUA or SBC.

Measurement Units:

NA

Issues:

None.

See Also:

Signaling Plane

3.2.3. SIP Transport Protocol

Definition:

The protocol used for transport of the Signaling Plane messages.

Discussion:

Performance benchmarks may vary for the same SIP networking device depending upon whether TCP, UDP, TLS, SCTP, websockets [RFC7118] or any future transport layer protocol is used. For this reason it is necessary to measure the SIP Performance Benchmarks using these various transport protocols. Performance Benchmarks MUST report the SIP Transport Protocol used to obtain the benchmark results.

Measurement Units:

While these are not units of measure, they are attributes that are one of many factors that will contribute to the value of the measurements to be taken. TCP, UDP, SCTP, TLS over TCP, TLS over UDP, TLS over SCTP, and websockets are among the possible values to be recorded as part of the test.

Issues:

None.

See Also:

None.

3.3. Test Setup Parameters**3.3.1. Session Attempt Rate****Definition:**

Configuration of the EA for the number of sessions per second (sps) that the EA attempts to establish using the services of the DUT.

Discussion:

The Session Attempt Rate is the number of sessions per second that the EA sends toward the DUT. Some of the sessions attempted may not result in a session being established.

Measurement Units:

Session attempts per second

Issues:

None.

See Also:

Session
Session Attempt

3.3.2. Establishment Threshold Time**Definition:**

Configuration of the EA that represents the amount of time that an EA client will wait for a response from an EA server before declaring a Session Attempt Failure.

Discussion:

This time duration is test dependent.

It is RECOMMENDED that the Establishment Threshold Time value be set to Timer B or Timer F as specified in RFC 3261, Table 4 [RFC3261].

Measurement Units:

Seconds

Issues:

None.

See Also:

None.

3.3.3. Session Duration**Definition:**

Configuration of the EA that represents the amount of time that the SIP dialog is intended to exist between the two EAs associated with the test.

Discussion:

The time at which the BYE is sent will control the Session Duration.

Measurement Units:

seconds

Issues:

None.

See Also:

None.

3.3.4. Media Packet Size**Definition:**

Configuration on the EA for a fixed number of frames or samples to be sent in each RTP packet of the media stream when the test involves Associated Media.

Discussion:

This document describes a method to measure SIP performance. If the DUT is processing media as well as SIP messages the media processing will potentially slow down the SIP processing and lower the SIP performance metric. The tests with associated media are designed for audio codecs and the assumption was made that larger media packets would require more processor time. This document does not define parameters applicable to video codecs.

For a single benchmark test, media sessions use a defined number of samples or frames per RTP packet. If two SBCs, for example, used the same codec but one puts more frames into the RTP packet, this might cause variation in the performance benchmark results.

Measurement Units:

An integer number of frames or samples, depending on whether hybrid- or sample-based codec are used, respectively.

Issues:

None.

See Also:

None.

3.3.5. Codec Type

Definition:

The name of the codec used to generate the media session.

Discussion

For a single benchmark test, all sessions use the same size packet for media streams. The size of packets can cause a variation in the performance benchmark measurements.

Measurement Units:

This is a textual name (alphanumeric) assigned to uniquely identify the codec.

Issues:

None.

See Also:

None.

3.4. Benchmarks

3.4.1. Session Establishment Rate

Definition:

The maximum value of the Session Attempt Rate that the DUT can handle for an extended, pre-defined, period with zero failures.

Discussion:

This benchmark is obtained with zero failure. The session attempt rate provisioned on the EA is raised and lowered as described in the algorithm in the accompanying methodology document [I-D.ietf-bmwg-sip-bench-meth], until a traffic load over the period of time necessary to attempt N sessions completes without failure, where N is a parameter specified in the algorithm and recorded in the Test Setup Report.

Measurement Units:

sessions per second (sps)

Issues:

None.

See Also:

Invite-Initiated Sessions
Non-Invite-Initiated Sessions
Session Attempt Rate

3.4.2. Registration Rate

Definition:

The maximum value of the Registration Attempt Rate that the DUT can handle for an extended, pre-defined, period with zero failures.

Discussion:

This benchmark is obtained with zero failures. The registration rate provisioned on the Emulated Agent is raised and lowered as described in the algorithm in the companion methodology draft [I-D.ietf-bmwg-sip-bench-meth], until a traffic load consisting of registration attempts at the given attempt rate over the period of time necessary to attempt N registrations completes without failure, where N is a parameter specified in the algorithm and recorded in the Test Setup Report.

This benchmark is described separately from the Session Establishment Rate (Section 3.4.1), although it could be considered a special case of that benchmark, since a REGISTER request is a request for a Non-Invite-Initiated session. It is defined separately because it is a very important benchmark for most SIP installations. An example demonstrating its use is an

avalanche restart, where hundreds of thousands of end points register simultaneously following a power outage. In such a case, an authoritative measurement of the capacity of the device to register endpoints is useful to the network designer. Additionally, in certain controlled networks, there appears to be a difference between the registration rate of new endpoints and the registering rate of existing endpoints (register refreshes). This benchmark can capture these differences as well.

Measurement Units:

registrations per second (rps)

Issues:

None.

See Also:

None.

3.4.3. Registration Attempt Rate

Definition:

Configuration of the EA for the number of registrations per second that the EA attempts to send to the DUT.

Discussion:

The Registration Attempt Rate is the number of registration requests per second that the EA sends toward the DUT.

Measurement Units:

Registrations per second (rps)

Issues:

None.

See Also: Non-Invite-Initiated Session

4. IANA Considerations

This document requires no IANA considerations.

5. Security Considerations

Documents of this type do not directly affect the security of Internet or corporate networks as long as benchmarking is not performed on devices or systems connected to production networks. Security threats and how to counter these in SIP and the media layer

is discussed in RFC3261 [RFC3261], RFC 3550 [RFC3550] and RFC3711 [RFC3711]. This document attempts to formalize a set of common terminology for benchmarking SIP networks. Packets with unintended and/or unauthorized DSCP or IP precedence values may present security issues. Determining the security consequences of such packets is out of scope for this document.

6. Acknowledgments

The authors would like to thank Keith Drage, Cullen Jennings, Daryl Malas, Al Morton, and Henning Schulzrinne for invaluable contributions to this document. Dale Worley provided an extensive review that lead to improvements in the documents. We are grateful to Barry Constantine, William Cervený and Robert Sparks for providing valuable comments during the document's last calls and expert reviews. Al Morton and Sarah Banks have been exemplary working group chairs, we thank them for tracking this work to completion.

7. References

7.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.
- [RFC5393] Sparks, R., Lawrence, S., Hawrylyshen, A., and B. Campen, "Addressing an Amplification Vulnerability in Session Initiation Protocol (SIP) Forking Proxies", RFC 5393, December 2008.
- [RFC4320] Sparks, R., "Actions Addressing Identified Issues with the Session Initiation Protocol's (SIP) Non-INVITE Transaction", RFC 4320, January 2006.
- [RFC6026] Sparks, R. and T. Zourzouvillys, "Correct Transaction Handling for 2xx Responses to Session Initiation Protocol (SIP) INVITE Requests", RFC 6026, September 2010.

[I-D.ietf-bmwg-sip-bench-meth]
Davids, C., Gurbani, V., and S. Poretsky, "SIP Performance Benchmarking Methodology",
draft-ietf-bmwg-sip-bench-meth-10 (work in progress),
May 2014.

7.2. Informational References

- [RFC2285] Mandeville, R., "Benchmarking Terminology for LAN Switching Devices", RFC 2285, February 1998.
- [RFC1242] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.
- [RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.
- [RFC6357] Hilt, V., Noel, E., Shen, C., and A. Abdelal, "Design Considerations for Session Initiation Protocol (SIP) Overload Control", RFC 6357, August 2011.
- [RFC7118] Baz Castillo, I., Millan Villegas, J., and V. Pascual, "The WebSocket Protocol as a Transport for the Session Initiation Protocol (SIP)", RFC 7118, January 2014.

Authors' Addresses

Carol Davids
Illinois Institute of Technology
201 East Loop Road
Wheaton, IL 60187
USA

Phone: +1 630 682 6024
Email: davids@iit.edu

Vijay K. Gurbani
Bell Laboratories, Alcatel-Lucent
1960 Lucent Lane
Rm 9C-533
Naperville, IL 60566
USA

Phone: +1 630 224 0216
Email: vkg@bell-labs.com

Scott Poretsky
Allot Communications
300 TradeCenter, Suite 4680
Woburn, MA 08101
USA

Phone: +1 508 309 2179
Email: sporetsky@allot.com

Benchmarking Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 22, 2012

R. Papneja
Huawei Technologies
B. Parise
Cisco Systems
S. Hares
Huawei Technologies
I. Varlashkin
Easynet Global Services
October 20, 2011

Basic BGP Convergence Benchmarking Methodology for Data Plane
Convergence
draft-papneja-bgp-basic-dp-convergence-02.txt

Abstract

BGP is widely deployed and used by several service providers as the default Inter AS routing protocol. It is of utmost importance to ensure that when a BGP peer or a downstream link of a BGP peer fails, the alternate paths are rapidly used and routes via these alternate paths are installed. This document provides the basic BGP Benchmarking Methodology using existing BGP Convergence Terminology, RFC 4098.

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 <http://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 April 22, 2012.

Copyright Notice

Copyright (c) 2011 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
(<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1. Introduction	4
1.1. Precise Benchmarking Definition	4
1.2. Purpose of BGP FIB (Data Plane) Convergence	5
1.3. Control Plane Convergence	5
1.4. Benchmarking Testing	5
2. Existing Definitions and Requirements	5
3. Test Topologies	6
3.1. General Reference Topologies	6
4. Test Considerations	8
4.1. Number of Peers	8
4.2. Number of Routes per Peer	8
4.3. Policy Processing/Reconfiguration	9
4.4. Configured Parameters (Timers, etc..)	9
4.5. Interface Types	10
4.6. Measurement Accuracy	10
4.7. Measurement Statistics	11
4.8. Authentication	11
4.9. Convergence Events	11
4.10. High Availability	11
5. Test Cases	12
5.1. Basic Convergence Tests	12
5.1.1. RIB-IN Convergence	12
5.1.2. RIB-OUT Convergence	13
5.1.3. eBGP Convergence	15
5.1.4. iBGP Convergence	15
5.1.5. eBGP Multihop Convergence	16
5.2. BGP Failure/Convergence Events	17
5.2.1. Physical Link Failure on DUT End	17
5.2.2. Physical Link Failure on Remote/Emulator End	18
5.2.3. ECMP Link Failure on DUT End	18
5.3. BGP Adjacency Failure (Non-Physical Link Failure) on Emulator	19
5.4. BGP Hard Reset Test Cases	20
5.4.1. BGP Non-Recovering Hard Reset Event on DUT	20
5.5. BGP Soft Reset	21
5.6. BGP Route Withdrawal Convergence Time	22
5.7. BGP Path Attribute Change Convergence Time	24
5.8. BGP Graceful Restart Convergence Time	26
6. Reporting Format	27
7. IANA Considerations	30
8. Security Considerations	30
9. Acknowledgments	30
10. References	30
10.1. Normative References	30
10.2. Informative References	31
Authors' Addresses	31

1. Introduction

This document defines the methodology for benchmarking data plane FIB convergence performance of BGP in router and switches for simple topologies of 3 or 4 nodes. The methodology proposed in this document applies to both IPv4 and IPv6 and if a particular test is unique to one version, it is marked accordingly. For IPv6 benchmarking the device under test will require the support of Multi-Protocol BGP (MP-BGP) [RFC4760, RFC2545].

The scope of this companion document is limited to basic BGP protocol FIB convergence measurements. BGP extensions outside of carrying IPv6 in (MP-BGP) [RFC4760, RFC2545] are outside the scope of this document. Interaction with IGP (IGP interworking) is outside the scope of this document.

This document makes assumption that ability of a router to forward packets implies that BGP has converged. Some modern routers may have an optimisation which results in forwarding ability is restored before BGP converges. Such optimisation **MUST** be disabled before applying benchmarking methodology described here. If that is not possible or if it's not known whether or not router has such optimisation alternative approach described in [draft-varlashkin-router-conv-bench] should be used instead.

1.1. Precise Benchmarking Definition

Since benchmarking is science of precision, let us restate the purpose of this document in benchmarking terms. This document defines methodology to test

- data plane convergence on a single BGP device that supports the BGP [RFC4271] functionality
- in test topology of 3 or 4 nodes
- using Basic BGP

Data plane convergence is defined as the completion of all FIB changes so that all forwarded traffic now takes the new proposed route. RFC 4098 defines the terms BGP device, FIB and the forwarded traffic. Data plane convergence is different than control plane convergence within a node.

Basic BGP is defined as RFC 4271 functional with Multi-Protocol BGP (MP-BGP) [RFC4760, RFC2545] for IPv6. The use of other extensions of BGP to support layer-2, layer-3 virtual private networks (VPN) are out of scope of this document.

The terminology used in this document is defined in [RFC4098]. One additional term is defined in this draft: FIB (Data plane) BGP Convergence.

1.2. Purpose of BGP FIB (Data Plane) Convergence

In the current Internet architecture the Inter-Autonomous System (inter-AS) transit is primarily available through BGP. To maintain a reliable connectivity within intra-domains or across inter-domains, fast recovery from failures remains most critical. To ensure minimal traffic losses, many service providers are requiring BGP implementations to converge the entire Internet routing table within sub-seconds at FIB level.

Furthermore, to compare these numbers amongst various devices, service providers are also looking at ways to standardize the convergence measurement methods. This document offers test methods for simple topologies. These simple tests will provide a quick high-level check, of the BGP data plane convergence across multiple implementations.

1.3. Control Plane Convergence

The convergence of BGP occurs at two levels: RIB and FIB convergence. RFC 4098 defines terms for BGP control plane convergence. Methodologies which test control plane convergence are out of scope for this draft.

1.4. Benchmarking Testing

In order to ensure that the results obtained in tests are repeatable, careful setup of initial conditions and exact steps are required.

This document proposes these initial conditions, test steps, and result checking. To ensure uniformity of the results all optional parameters SHOULD be disabled and all settings SHOULD be changed to default, these may include BGP timers as well.

2. Existing Definitions and Requirements

RFC 1242, "Benchmarking Terminology for Network Interconnect Devices" [RFC1242] and RFC 2285, "Benchmarking Terminology for LAN Switching Devices" [RFC2285] SHOULD be reviewed in conjunction with this document. WLAN-specific terms and definitions are also provided in Clauses 3 and 4 of the IEEE 802.11 standard [802.11]. Commonly used terms may also be found in RFC 1983 [RFC1983].

For the sake of clarity and continuity, this document adopts the general template for benchmarking terminology set out in Section 2 of RFC 1242. Definitions are organized in alphabetical order, and grouped into sections for ease of reference. The following terms are assumed to be taken as defined in RFC 1242 [RFC1242]: Throughput, Latency, Constant Load, Frame Loss Rate, and Overhead Behavior. In addition, the following terms are taken as defined in [RFC2285]: Forwarding Rates, Maximum Forwarding Rate, Loads, Device Under Test (DUT), and System Under Test (SUT).

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Test Topologies

This section describes simple test setups for use in BGP benchmarking tests measuring convergence of the FIB (data plane) after the BGP updates has been received.

These simple test nodes have 3 or 4 nodes with the following configuration:

1. Basic Test Setup
2. Three node setup for iBGP or eBGP convergence
3. Setup for eBGP multihop test scenario
4. Four node setup for iBGP or eBGP convergence

Individual tests refer to these topologies.

Figures 1-4 use the following conventions

- o AS-X: Autonomous System X
- o Loopback Int: Loopback interface on the BGP enabled device
- o R2: Helper router

3.1. General Reference Topologies

Emulator acts as 1 or more BGP peers for different testcases.

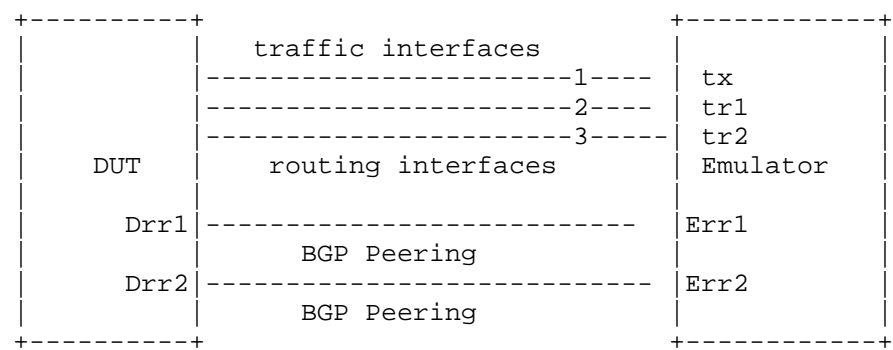


Figure 1 Basic Test Setup

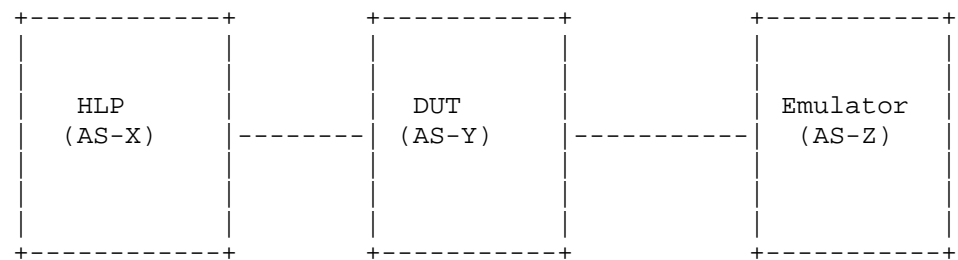


Figure 2 Three Node Setup for eBGP and iBGP Convergence

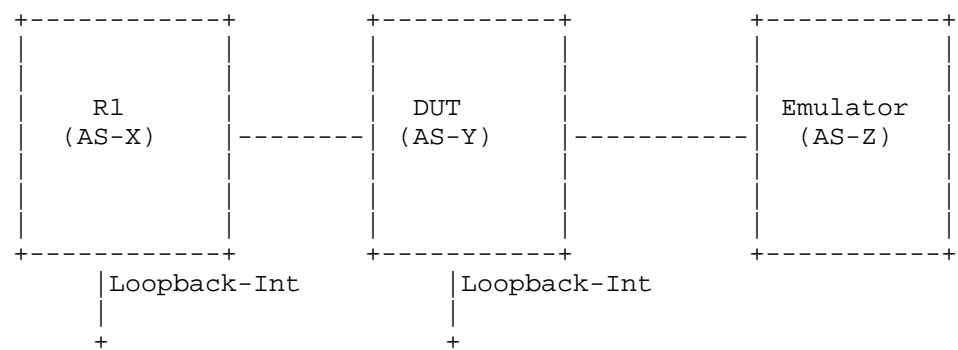


Figure 3 BGP Convergence for eBGP Multihop Scenario

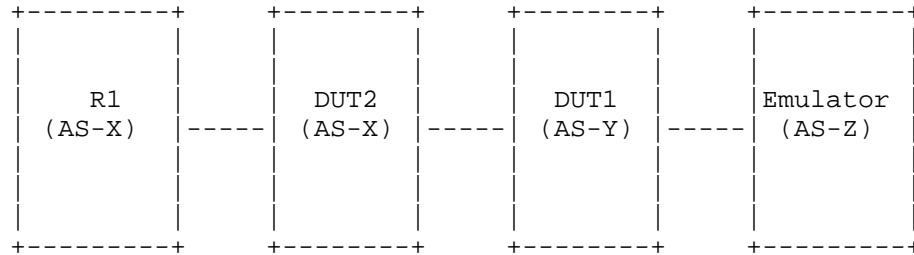


Figure 4 Four Node Setup for EBGP and IBGP Convergence

4. Test Considerations

The test cases for measuring convergence for iBGP and eBGP are different. Both iBGP and eBGP use different mechanisms to advertise, install and learn the routes. Typically, an iBGP route on the DUT is installed and exported only when the next-hop is reachable. For eBGP the route is installed on the DUT with the remote interface address as the next-hop with the exception of the multihop case.

4.1. Number of Peers

Number of Peers is defined as the number of BGP neighbors or sessions the DUT has at the beginning of the test. The peers are established before the tests begin. The relationship could be either, iBGP or eBGP peering depending upon the test case requirement.

The DUT establishes one or more BGP sessions with one more emulated routers or helper nodes. Additional peers can be added based on the testing requirements. The number of peers enabled during the testing should be well documented in the report matrix.

4.2. Number of Routes per Peer

It Number of Routes per Peer is defined as the number of routes advertized or learnt by the DUT per session or through neighbor relationship with an emulator or helper node. The tester, emulating as neighbor MUST advertise at least one route per peer.

Each test must run must identify the route stream in terms of route packing, route mixture, and number of routes. This route stream must be well documented in the reporting stream. RFC 4098 defines these terms.

It is RECOMMENDED that the user may consider advertizing the entire current Internet routing table per peering session using an Internet route mixture with unique or non-unique routes. If multiple peers are used, it is important to precisely document the timing sequence between the peer sending routes (as defined in RFC 4098).

4.3. Policy Processing/Reconfiguration

The DUT MUST run one baseline test where policy is Minimum policy as defined in RFC 4098. Additional runs may be done with policy set-up before the tests begin. Exact policy settings should be documented as part of the test.

4.4. Configured Parameters (Timers, etc..)

There are configured parameters and timers that may impact the measured BGP convergence times.

The benchmark metrics MAY be measured at any fixed values for these configured parameters.

It is RECOMMENDED these configure parameters have two settings: a) basic-test, and b) values as expected in the operational network. All optional BGP settings MUST be kept consistent across iterations of any specific tests

Examples of the configured parameters that may impact measured BGP convergence time include, but are not limited to:

1. Interface failure detection timer
2. BGP Keepalive timer
3. BGP Holdtime
4. BGP update delay timer
5. ConnectRetry timer
6. TCP Segment Size

7. Minimum Route Advertisement Interval (MRAI)
8. MinASOriginationInterval (MAOI)
9. Route Flap Dampening parameters
10. TCP MD5

The basic-test settings for the parameters should be:

1. Interface failure detection timer (0 ms)
2. BGP Keepalive timer (1 min)
3. BGP Holdtime (3 min)
4. BGP update delay timer (0 s)
5. ConnectRetry timer (1 s)
6. TCP Segment Size (4096)
7. Minimum Route Advertisement Interval (MRAI) (0 s)
8. MinASOriginationInterval (MAOI)(0 s)
9. Route Flap Dampening parameters (off)
10. TCP MD5 (off)

4.5. Interface Types

The type of media dictate which test cases may be executed, each interface type has unique mechanism for detecting link failures and the speed at which that mechanism operates will influence the measurement results. All interfaces MUST be of the same media and throughput for each test case.

4.6. Measurement Accuracy

Since observed packet loss is used to measure the route convergence time, the time between two successive packets offered to each individual route is the highest possible accuracy of any packet-loss based measurement. When packet jitter is much less than the convergence time, it is a negligible source of error and hence it will be treated as within tolerance.

An exterior measurement on the input media (such Ethernet) is defined by this specification.

4.7. Measurement Statistics

The benchmark measurements may vary for each trial, due to the statistical nature of timer expirations, CPU scheduling, etc. It is recommended to repeat the test multiple times. Evaluation of the test data must be done with an understanding of generally accepted testing practices regarding repeatability, variance and statistical significance of a small number of trials.

For any repeated tests that are averaged to remove variance, all parameters MUST remain the same.

4.8. Authentication

Authentication in BGP is done using the TCP MD5 Signature Option [RFC5925]. The processing of the MD5 hash, particularly in devices with a large number of BGP peers and a large amount of update traffic, can have an impact on the control plane of the device. If authentication is enabled, it SHOULD be documented correctly in the reporting format

4.9. Convergence Events

Convergence events or triggers are defined as abnormal occurrences in the network, which initiate route flapping in the network, and hence forces the re-convergence of a steady state network. In a real network, a series of convergence events may cause convergence latency operators desire to test.

These convergence events must be defined in terms of the sequences defined in RFC 4098. This basic document begins all tests with a router initial set-up. Additional documents will define BGP data plane convergence based on peer initialization.

The convergence events may or may not be tied to the actual failure A Soft Reset (RFC 4098) does not clear the RIB or FIB tables. A Hard reset clears the BGP peer sessions, the RIB tables, and FIB tables.

4.10. High Availability

Due to the different Non-Stop-Routing (sometimes referred to High-Availability) solutions available from different vendors, it is RECOMMENDED that any redundancy available in the routing processors should be disabled during the convergence measurements.

5. Test Cases

All tests defined under this section assume the following:

- a. BGP peers should be brought to BGP Peer established state
- b. Furthermore the traffic generation and routing should be verified in the topology

5.1. Basic Convergence Tests

These test cases measure characteristics of a BGP implementation in non-failure scenarios like:

1. RIB-IN Convergence
2. RIB-OUT Convergence
3. eBGP Convergence
4. iBGP Convergence

5.1.1. RIB-IN Convergence

Objective:

This test measures the convergence time taken to receive and install a route in RIB using BGP

Reference Test Setup:

This test uses the setup as shown in figure 1

Procedure:

- A. All variables affecting Convergence should be set to a basic test state (as defined in section 4-4).
- B. Establish BGP adjacency between DUT and peer x of Emulator.
- C. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the

rest of the test.

- D. Start the traffic from the Emulator peer-x towards the DUT targeted at a routes specified in route mixture (ex. route A) Initially no traffic SHOULD be observed on the egress interface as the route A is not installed in the forwarding database of the DUT.

- E. Advertise route A from the Peer-x to the DUT and record the time.

This is $Tup(EMx, Rt-A)$ also named 'XMT-Rt-time'.

- F. Record the time when the route-A from Peer-x is received at the DUT.

This $Tup(DUT, Rt-A)$ also named 'RCV-Rt-time'.

- G. Record the time when the traffic targeted towards route A is received by Emulator on appropriate traffic egress interface.

This is $TR(TDx, Rt-A)$. This is also named DUT-XMT-Data-Time.

- H. The difference between the $Tup(TDx, RT-A)$ and traffic received time ($TR(TDr, Rt-A)$) is the FIB Convergence Time for route-A in the route mixture. A full convergence for the route update is the measurement between the 1st route (Route-A) and the last route (Rt-last)

Route update convergence is

$TR(TDr, RT-last) - Tup(DUT, Rt-A)$ or

$(DUT-XMT-Data-Time - RCV-Rt-Time)(rt-A)$

Note: It is recommended that a single test with the same route mixture be repeated several times. A report should provide the Standard Deviation of all tests and the Average.

Running tests with a varying number of routes and route mixtures is important to get a full characterization of a single peer.

5.1.2. RIB-OUT Convergence

Objective:

This test measures the convergence time taken by an implementation to receive, install and advertise a route using BGP

Reference Test Setup:

This test uses the setup as shown in figure 2

Procedure:

- A. The Helper node (HLP) run same version of BGP as DUT.
- B. All devices MUST be synchronized using NTP or some local reference clock.
- C. All configuration variables for HLP, DUT, and Emulator SHOULD be set to the same values. These values MAY be basic-test or a unique set completely described in the test set-up.
- D. Establish BGP adjacency between DUT and Emulator.
- E. Establish BGP adjacency between DUT and Helper Node.
- F. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the rest of the test
- G. Start the traffic from the Emulator towards the Helper Node targeted at a specific route say route A. Initially no traffic SHOULD be observed on the egress interface as the route-A is not installed in the forwarding database of the DUT.
- H. Advertise routeA from the Emulator to the DUT and note the time.

This is Tup(EMx, Route-A). (also named EM-XMT-Rt-Time

- I. Record when Route-A is received by DUT.

This is Tup(DUTr, Route-A). (also named DUT-RCV-Rt-Time)

- J. Record the time when the ROUTE forward by DUT toward the Helper node.

This is Tup(DUTx, Rt-A). (also named DUT-XMT-Rt-Time)

- K. Record the time when the traffic targeted towards route-A is received on the Route Egress Interface toward peer-X. This is TR(EMr, Route-A). (also named DUT-XMT-Data Time).

FIB convergence = (DUT-RCV-Rt-Time - DUT-XMT-Data-Time)

RIB convergence = (DUT-RCV-Rt-Time - DUT-XMT-Rt-Time)

Convergence for a route stream is characterized by

a) Individual route convergence for FIB, RIB

b) All route convergence of

FIB-convergence = DUT-RCV-Rt-Time(A) - DUT-XMT-Data-Time(last)

RIB-convergence = DUT-RCV-Rt-Time(A) - DUT-XMT-Rt-Time(last)

5.1.3. eBGP Convergence

Objective:

This test measures the convergence time taken by an implementation to receive, install and advertise a route in an eBGP Scenario

Reference Test Setup:

This test uses the setup as shown in figure 2 and the scenarios described in RIB-IN and RIB-OUT are applicable to this test case.

5.1.4. iBGP Convergence

Objective:

This test measures the convergence time taken by an implementation to receive, install and advertise a route in an iBGP Scenario

Reference Test Setup:

This test uses the setup as shown in figure 2 and the scenarios described in RIB-IN and RIB-OUT are applicable to this test case.

5.1.5. eBGP Multihop Convergence

Objective:

This test measures the convergence time taken by an implementation to receive, install and advertise a route in an eBGP Multihop Scenario

Reference Test Setup:

This test uses the setup as shown in figure 3. Two DUTs are used along with a helper node.

Procedure:

- A. The DUT2 is the same model as DUT and runs the same BGP implementation as DUT
- B. All devices to be synchronized using NTP
- C. All variables affecting Convergence like authentication, policies, timers should be set to basic-settings
- D. All 3 devices, DUT, Emulator and Helper Node are configured as different Autonomous Systems
- E. Loopback Interfaces configured on DUT and Helper Node and connectivity is established between them using any config options available on the DUT
- F. Establish BGP adjacency between DUT1 and Emulator
- G. Establish BGP adjacency between DUT2 and Helper Node
- H. Establish BGP adjacency between DUT 1 and DUT 2
- I. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT1 and DUT2 or a configurable delay before proceeding with the rest of the test
- J. Start the traffic from the Emulator towards the Helper Node targeted at a specific route say routeA
- K. Initially no traffic SHOULD be observed on the egress interface as the routeA is not installed in the forwarding database of the DUT

- L. Advertise routeA from the Emulator to the DUT and note the time. (Tup(EMx,RouteA) - This is also named (Route-Rec-time)
- M. Record the time when the traffic targeted towards routeA is received from Egress Interface of DUT on emulator This is TR(EMr,DUT), nicknamed (Data Receive time)
- N. The following equation represents the FIB Convergence multi-node

$$\text{eBGP Multihop Convergence Time} = (\text{Rt-RecTime} - \text{Data-RcvTime})$$

Note: It is recommended that the test be repeated with varying number of routes and route mixtures. With each set route mixture, the test should be repeated multiple times. The results should record average, mean, Standard Deviation

5.2. BGP Failure/Convergence Events

5.2.1. Physical Link Failure on DUT End

Objective:

This test measures the route convergence time due to local link failure event at DUT's Local Interface

Reference Test Setup:

This test uses the setup as shown in figure 1. Shutdown event is defined as an administrative shutdown event on the DUT

Procedure:

- A. All variables affecting Convergence like authentication, policies, timers should be set to basic-test policy
- B. Establish 2 BGP adjacencies from DUT to Emulator, one over the peer interface and the other using a second peer interface
- C. Advertise the same route, route A over both the adjacencies and (Tx1)Interface to be the preferred next hop
- D. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the rest of the test

- E. Start the traffic from the Emulator towards the DUT targeted at a specific route say route A. Initially traffic would be observed on the best egress route (Err1) instead of Trr2
- F. Trigger the shutdown event of Best Egress Interface on DUT (Drr1)
- G. Measure the Convergence Time for the event to be detected and traffic to be forwarded to Next-Best Egress Interface (rr2)

Time = Data-detect(rr2) - Shutdown time
- H. Stop the offered load and wait for the queues to drain and Restart
- I. Bring up the link on DUT Best Egress Interface
- J. Measure the convergence time taken for the traffic to be rerouted from (rr2) to Best Interface (rr1)

Time = Data-detect(rr1) - Shutdown time
- K. It is recommended that the test be repeated with varying number of routes and route mixtures or with number of routes and route mixtures closer to what is deployed in operational networks

5.2.2. Physical Link Failure on Remote/Emulator End

Objective:

This test measures the route convergence time due to local link failure event at Tester's Local Interface

Reference Test Setup:

This test uses the setup as shown in figure 1. Shutdown event is defined as shutdown of the local interface of Tester via logical shutdown event. The procedure used in 5.2.1 is used for the termination

5.2.3. ECMP Link Failure on DUT End

Objective:

This test measures the route convergence time due to local link failure event at ECMP Member. The FIB configuration and BGP is set to allow two ECMP routes to be installed. However, policy

directs the routes to be sent only over one of the paths

Reference Test Setup:

This test uses the setup as shown in figure 1 and the procedure uses 5.2.1

5.3. BGP Adjacency Failure (Non-Physical Link Failure) on Emulator

Objective:

This test measures the route convergence time due to BGP Adjacency Failure on Emulator

Reference Test Setup:

This test uses the setup as shown in figure 1

Procedure:

- A. All variables affecting Convergence like authentication, policies, timers should be basic-policy set
- B. Establish 2 BGP adjacencies from DUT to Emulator, one over the Best Egress Interface and the other using the Next-Best Egress Interface
- C. Advertise the same route, routeA over both the adjacencies and make Best Egress Interface to be the preferred next hop
- D. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the rest of the test
- E. Start the traffic from the Emulator towards the DUT targeted at a specific route say routeA. Initially traffic would be observed on the Best Egress interface
- F. Remove BGP adjacency via a software adjacency down on the Emulator on the Best Egress Interface. This time is called BGPAdj-down-time also termed BGPpeer-down
- G. Measure the Convergence Time for the event to be detected and traffic to be forwarded to Next-Best Egress Interface. This time is Tr-rr2 also called TR2-traffic-on

Convergence = TR2-traffic-on - BGPpeer-down

- H. Stop the offered load and wait for the queues to drain and Restart
- I. Bring up BGP adjacency on the Emulator over the Best Egress Interface. This time is BGP-adj-up also called BGPpeer-up
- J. Measure the convergence time taken for the traffic to be rerouted to Best Interface. This time is BGP-adj-up also called BGPpeer-up

5.4. BGP Hard Reset Test Cases

5.4.1. BGP Non-Recovering Hard Reset Event on DUT

Objective:

This test measures the route convergence time due to Hard Reset on the DUT

Reference Test Setup:

This test uses the setup as shown in figure 1

Procedure:

- A. The requirement for this test case is that the Hard Reset Event should be non-recovering and should affect only the adjacency between DUT and Emulator on the Best Egress Interface
- B. All variables affecting SHOULD be set to basic-test values
- C. Establish 2 BGP adjacencies from DUT to Emulator, one over the Best Egress Interface and the other using the Next-Best Egress Interface
- D. Advertise the same route, routeA over both the adjacencies and make Best Egress Interface to be the preferred next hop
- E. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the rest of the test

- F. Start the traffic from the Emulator towards the DUT targeted at a specific route say routeA. Initially traffic would be observed on the Best Egress interface
- G. Trigger the Hard Reset event of Best Egress Interface on DUT
- H. Measure the Convergence Time for the event to be detected and traffic to be forwarded to Next-Best Egress Interface

Time of convergence = time-traffic flow - time-reset

- I. Stop the offered load and wait for the queues to drain and Restart
- J. It is recommended that the test be repeated with varying number of routes and route mixtures or with number of routes and route mixtures closer to what is deployed in operational networks
- K. When varying number of routes are used, convergence Time is measured using the Loss Derived method [IGPData]
- L. Convergence Time in this scenario is influenced by Failure detection time on Tester, BGP Keep Alive Time and routing, forwarding table update time

5.5. BGP Soft Reset

Objective:

This test measures the route convergence time taken by an implementation to service a BGP Route Refresh message and advertise a route

Reference Test Setup:

This test uses the setup as shown in figure 2

Procedure:

- A. The BGP implementation on DUT and Helper Node needs to support BGP Route Refresh Capability [RFC2918]
- B. All devices to be synchronized using NTP

- C. All variables affecting Convergence like authentication, policies, timers should be set to basic-test defaults
- D. DUT and Helper Node are configured in the same Autonomous System whereas Emulator is configured under a different Autonomous System
- E. Establish BGP adjacency between DUT and Emulator
- F. Establish BGP adjacency between DUT and Helper Node
- G. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the rest of the test
- H. Configure a policy under BGP on Helper Node to deny routes received from DUT
- I. Advertise routeA from the Emulator to the DUT
- J. The DUT will try to advertise the route to Helper Node will be denied
- K. Wait for 3 KeepAlives
- L. Start the traffic from the Emulator towards the Helper Node targeted at a specific route say routeA. Initially no traffic would be observed on the Egress interface, as routeA is not present
- M. Remove the policy on Helper Node and issue a Route Refresh request towards DUT. Note the timestamp of this event. This is the RefreshTime
- N. Record the time when the traffic targeted towards routeA is received on the Egress Interface. This is RecTime
- O. The following equation represents the Route Refresh Convergence Time per route

$$\text{Route Refresh Convergence Time} = (\text{RecTime} - \text{RefreshTime})$$

5.6. BGP Route Withdrawal Convergence Time

Objective:

This test measures the route convergence time taken by an implementation to service a BGP Withdraw message and advertise the withdraw

Reference Test Setup:

This test uses the setup as shown in figure 2

Procedure:

- A. This test consists of 2 steps to determine the Total Withdraw Processing Time
- B. Step 1:
 - (1) All devices to be synchronized using NTP
 - (2) All variables should be set to basic-test parameters
 - (3) DUT and Helper Node are configured in the same Autonomous System whereas Emulator is configured under a different Autonomous System
 - (4) Establish BGP adjacency between DUT and Emulator
 - (5) To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the rest of the test
 - (6) Start the traffic from the Emulator towards the DUT targeted at a specific route say routeA. Initially no traffic would be observed on the Egress interface as the routeA is not present on DUT
 - (7) Advertise routeA from the Emulator to the DUT
 - (8) The traffic targeted towards routeA is received on the Egress Interface
 - (9) Now the Tester sends request to withdraw routeA to DUT, TRx(Awith) also called WdrawTime1
 - (10) Record the time when no traffic is observed on the Egress Interface. This is the RouteRemoveTime1(A)

$WdrawConvTime1 = RouteRemoveTime1(A)$

- (11) The difference between the `RouteRemoveTime1` and `WdrawTime1` is the `WdrawConvTime1`

C. Step 2:

- (1) Continuing from Step 1, re-advertise routeA back to DUT from Tester
- (2) The DUT will try to advertise the routeA to Helper Node (assumption there exists a session between DUT and helper node)
- (3) Start the traffic from the Emulator towards the Helper Node targeted at a specific route say routeA. Traffic would be observed on the Egress interface after routeA is received by the Helper Node

$WATime = \text{time traffic first flows}$

- (4) Now the Tester sends a request to withdraw routeA to DUT. This is the `WdrawTime2`

$WAWtime - TRx(RouteA) = WdrawTime2$

- (5) DUT processes the withdraw and sends it to Helper Node
- (6) Record the time when no traffic is observed on the Egress Interface of Helper Node. This is

$TR - WAW(DUT, RouteA) = RouteRemoveTime2$

- (7) Total withdraw processing time is

$TotalWdrawTime = ((RouteRemoveTime2 - WdrawTime2) - WdrawConvTime1)$

5.7. BGP Path Attribute Change Convergence Time

Objective:

This test measures the convergence time taken by an implementation to service a BGP Path Attribute Change

Reference Test Setup:

This test uses the setup as shown in figure 1

Procedure:

- A. This test only applies to Well-Known Mandatory Attributes like Origin, AS Path, Next Hop
- B. In each iteration of test only one of these mandatory attributes need to be varied whereas the others remain the same
- C. All devices to be synchronized using NTP
- D. All variables should be set to basic-test parameters
- E. Advertise the route, routeA over the Best Egress Interface only, making it the preferred next hop
- F. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the rest of the test
- G. Start the traffic from the Emulator towards the DUT targeted at the specific route say routeA. Initially traffic would be observed on the Best Egress interface
- H. Now advertise the same route routeA on the Next-Best Egress Interface but by varying one of the well-known mandatory attributes to have a preferred value over that interface. The other values need to be same as what was advertised on the Best-Egress adjacency

$TRx(\text{Path-Change}) = \text{Path Change Event Time}$

- I. Measure the Convergence Time for the event to be detected and traffic to be forwarded to Next-Best Egress Interface

$DUT(\text{Path-Change, RouteA}) = \text{Path-switch time}$

$\text{Convergence} = \text{Path-switch time} - \text{Path Change Event Time}$

- J. Stop the offered load and wait for the queues to drain and Restart

5.8. BGP Graceful Restart Convergence Time

Objective:

This test measures the route convergence time taken by an implementation during a Graceful Restart Event

Reference Test Setup:

This test uses the setup as shown in figure 4

Procedure:

- A. It measures the time taken by an implementation to service a BGP Graceful Restart Event and advertise a route
- B. The Helper Nodes are the same model as DUT and run the same BGP implementation as DUT
- C. The BGP implementation on DUT and Helper Node needs to support BGP Graceful Restart Mechanism [RFC4724]
- D. All devices to be synchronized using NTP
- E. All variables are set to basic-test values
- F. DUT and Helper Node-1 are configured in the same Autonomous System whereas Emulator and Helper Node-2 are configured under different Autonomous Systems
- G. Establish BGP adjacency between DUT and Helper Nodes
- H. Establish BGP adjacency between Helper Node-2 and Emulator
- I. To ensure adjacency establishment, wait for 3 KeepAlives from the DUT or a configurable delay before proceeding with the rest of the test
- J. Configure a policy under BGP on Helper Node-1 to deny routes received from DUT
- K. Advertise routeA from the Emulator to Helper Node-2
- L. Helper Node-2 advertises the route to DUT and DUT will try to advertise the route to Helper Node-1 which will be denied

- M. Wait for 3 KeepAlives
- N. Start the traffic from the Emulator towards the Helper Node-1 targeted at the specific route say routeA. Initially no traffic would be observed on the Egress interface as the routeA is not present
- O. Perform a Graceful Restart Trigger Event on DUT and note the time. This is the GREventTime
- P. Remove the policy on Helper Node-1
- Q. Record the time when the traffic targeted towards routeA is received on the Egress Interface

TRr(DUT, routeA). This is also called RecTime
- R. The following equation represents the Graceful Restart Convergence Time

$$\text{Graceful Restart Convergence Time} = ((\text{GREventTime} - \text{RecTime}) - \text{RIB-IN})$$
- S. It is assumed in this test case that after a Switchover is triggered on the DUT, it will not have any cycles to process BGP Refresh messages. The reason for this assumption is that there is a narrow window of time where after switchover when we remove the policy from Helper Node -1, implementations might generate Route-Refresh automatically and this request might be serviced before the DUT actually switches over and reestablishes BGP adjacencies with the peers

6. Reporting Format

For each test case, it is recommended that the reporting tables below are completed and all time values SHOULD be reported with resolution as specified in [RFC4098]

Parameter	Units
Test case	Test case number
Test topology	1,2,3 or 4
Parallel links	Number of parallel links
Interface type	GigE, POS, ATM, other
Convergence Event	Hard reset, Soft reset, link failure, or other defined
eBGP sessions	Number of eBGP sessions
iBGP sessions	Number of iBGP sessions
eBGP neighbor	Number of eBGP neighbors
iBGP neighbor	Number of iBGP neighbors
Routes per peer	Number of routes
Total unique routes	Number of routes
Total non-unique routes	Number of routes
IGP configured	ISIS, OSPF, static, or other
Route Mixture	Description of Route mixture
Route Packing	Number of routes in an update
Policy configured	Yes, No
Packet size offered to the DUT	Bytes
Offered load	Packets per second
Packet sampling interval on tester	Seconds
Forwarding delay threshold	Seconds
Timer Values configured on DUT	
Interface failure indication delay	Seconds
Hold time	Seconds
MinRouteAdvertisementInterval (MRAI)	Seconds
MinASOriginationInterval (MAOI)	Seconds
Keepalive Time	Seconds
ConnectRetry	Seconds
TCP Parameters for DUT and tester	
MSS	Bytes
Slow start threshold	Bytes
Maximum window size	Bytes

Test Details:

- a. If the Offered Load matches a subset of routes, describe how this subset is selected
- b. Describe how the Convergence Event is applied; does it cause instantaneous traffic loss or not

c. If there is any policy configured, describe the configured policy

Complete the table below for the initial Convergence Event and the reversion Convergence Event

Parameter	Unit
Convergence Event	Initial or reversion
Traffic Forwarding Metrics	
Total number of packets offered to DUT	Number of packets
Total number of packets forwarded by DUT	Number of packets
Connectivity Packet Loss	Number of packets
Convergence Packet Loss	Number of packets
Out-of-order packets	Number of packets
Duplicate packets	Number of packets
Convergence Benchmarks	
Rate-derived Method[IGP-Data]:	
First route convergence time	Seconds
Full convergence time	Seconds
Loss-derived Method [IGP-Data]:	
Loss-derived convergence time	Seconds
Route-Specific Loss-Derived Method:	
Minimum R-S convergence time	Seconds
Maximum R-S convergence time	Seconds
Median R-S convergence time	Seconds
Average R-S convergence time	Seconds
Loss of Connectivity Benchmarks	
Loss-derived Method:	
Loss-derived loss of connectivity period	Seconds
Route-Specific loss-derived Method:	
Minimum LoC period [n]	Array of seconds
Minimum Route LoC period	Seconds
Maximum Route LoC period	Seconds
Median Route LoC period	Seconds

Average Route LoC period Seconds

7. IANA Considerations

This draft does not require any new allocations by IANA.

8. Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the constraints specified in the sections above.

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT/SUT.

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

9. Acknowledgments

The following people made textual contributions to this document: Jay Karthik (Cisco Systems), Eric Brendel (AT&T), and Mohan Nanduri (Microsoft) The authors would like to thank them for the helpful discussions and their contributions to the document.

10. References

10.1. Normative References

[I-D.ietf-bmwg-igp-dataplane-conv-term]
Poretsky, S., Imhoff, B., and K. Michielsen, "Terminology for Benchmarking Link-State IGP Data Plane Route Convergence", draft-ietf-bmwg-igp-dataplane-conv-term-23 (work in progress), February 2011.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2918] Chen, E., "Route Refresh Capability for BGP-4", RFC 2918, September 2000.
- [RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

10.2. Informative References

- [RFC1242] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [RFC1983] Malkin, G., "Internet Users' Glossary", RFC 1983, August 1996.
- [RFC2285] Mandeville, R., "Benchmarking Terminology for LAN Switching Devices", RFC 2285, February 1998.
- [RFC2545] Marques, P. and F. Dupont, "Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing", RFC 2545, March 1999.
- [RFC4098] Berkowitz, H., Davies, E., Hares, S., Krishnaswamy, P., and M. Lepp, "Terminology for Benchmarking BGP Device Convergence in the Control Plane", RFC 4098, June 2005.
- [RFC4724] Sangli, S., Chen, E., Fernando, R., Scudder, J., and Y. Rekhter, "Graceful Restart Mechanism for BGP", RFC 4724, January 2007.
- [RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, January 2007.
- [RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, June 2010.

Authors' Addresses

Rajiv Papneja
Huawei Technologies

Email: rajiv.papneja@huawei.com

Bhavani Parise
Cisco Systems

Email: bhavani@cisco.com

Susan Hares
Huawei Technologies

Email: shares@huawei.com

Ilya Varlashkin
Easynet Global Services

Email: ilya.varlashkin@easynet.com

Eric Brendel
Independent Consultant

Email: brendel@pektel.com

Mohan Nanduri
Microsoft

Email: mnanduri@microsoft.com

Jay Karthik
Cisco Systems

Email: jkarthik@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 22, 2011

T. Player
Spirent Communications
D. Newman
Network Test
October 19, 2010

Bridge Out: Benchmarking Methodology Extensions for Data Center Bridging
Devices
draft-player-dcb-benchmarking-03.txt

Abstract

Existing benchmarking methodologies are based on the assumption that networking devices will impartially drop network traffic at their performance limits. Data Center Bridging (DCB) devices, however, will attempt to throttle prioritized traffic from network endpoints before those limits are reached in order to minimize the probability of frame loss for high value traffic. Hence, existing methodologies based around indiscriminate frame loss are inappropriate for DCB devices. This document takes the basic benchmarking ideas based on loss and extends them to support "lossless" Ethernet devices.

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 <http://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 April 22, 2011.

Copyright Notice

Copyright (c) 2010 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	4
2. Requirements	4
3. Terminology	4
4. General Considerations	5
4.1. Classifications	5
4.2. Congestion	5
4.3. Test Traffic	6
4.4. Tester Capabilities	6
4.4.1. Frame Formats	6
4.4.2. Pause Response Time	7
5. Test Setup	7
5.1. Test Traffic	7
5.1.1. Traffic Classification	7
5.1.2. Trial Duration	7
5.1.3. Frame Measurements	7
5.1.4. Frame Sizes	8
5.1.5. Burst Sizes	8
6. Benchmarking Tests	9
6.1. Pause Response Time	9
6.1.1. Objective	9
6.1.2. Setup Parameters	9
6.1.3. Procedure	10
6.1.4. Measurements	10
6.1.5. Reporting Format	11
6.2. Queueput	11
6.2.1. Objective	11
6.2.2. Setup Parameters	11
6.2.3. Procedure	11
6.2.4. Measurements	12
6.2.5. Reporting Format	12
6.3. Maximum Forwarding Rate	12
6.3.1. Objective	12
6.3.2. Setup Parameters	12
6.3.3. Procedure	13
6.3.4. Measurements	13
6.3.5. Reporting Format	14
6.4. Back-off	14
6.4.1. Objective	14

6.4.2. Setup Parameters	14
6.4.3. Procedure	15
6.4.4. Measurements	15
6.4.5. Reporting Format	15
6.5. Back-to-Back	15
6.5.1. Objective	15
6.5.2. Setup Parameters	15
6.5.3. Procedure	16
6.5.4. Measurements	16
6.5.5. Reporting Format	17
7. Security Considerations	17
8. IANA Consdierations	17
9. Normative References	17
Appendix A. Acknowledgements	18
Authors' Addresses	18

1. Introduction

This document is intended to provide a methodology for benchmarking Data Center Bridging (DCB) devices that support Priority-based Flow Control (PFC). It extends the methodologies already defined in [RFC2544] and [RFC2889].

This memo primarily deals with devices which use Priority-based Flow Control, as defined in IEEE specification 802.1Qbb, to actively manage the transmission rate of multiple classes of traffic in order to minimize forwarding delay and frame loss for high priority traffic.

2. Requirements

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].

3. Terminology

As the terminology used by [RFC4689] is specific to IP layer testing, a number of existing terms require clarification when used in the DCB benchmarking context. Additionally, a number of new terms are also presented to clarify concepts not clearly defined within the scope of [RFC4689].

Classification: As stated in [RFC4689], Classification is the selection of packets according to defined rules. In the context of DCB benchmarking, the Classification criterion is the value of the 802.1p priority code point field in the 802.1Q VLAN header of an Ethernet frame.

Classification Group: A collection of traffic streams that belong to a single Classification. A Conformance Vector MAY be associated with a Classification Group.

Classification Profile: The set of all Classification Groups involved in a benchmarking test.

Conformance Vector: A set of measurable stream result bounds, e.g. latency, jitter, sequencing, etc., that specify whether a frame is Conformant or Non-conformant. Conformance vectors are optional for all DCB benchmarking tests.

Congestion Management: In the context of DCB benchmarking, Congestion Management occurs when the DUT/SUT transmits Priority-based Flow Control (PFC) Pause frames.

Forwarding Congestion: In the context of DCB benchmarking, Forwarding Congestion is extended to include the observation of PFC pause frame transmissions from the DUT.

Intended Load: In this document, the Intended Load refers to the summation of the Intended Vectors for all Classification Groups.

Offered Load: In this document, the Offered Load refers to the summation of the Offered Vectors for all Classification Groups.

Queue Congestion: Queue congestion occurs when a DUT/SUT uses Congestion Management on a set of traffic Classifications. The congestion Classifications correspond to the congested queues in the DUT/SUT.

Queueput: The maximum Offered Load that can be transmitted into a DUT/SUT such that every transmitted frame matches a specific Classification rule, the DUT/SUT does NOT use priority-based flow control mechanisms to manage the ingress traffic rate of the Classification(s) of interest, and all ingress frames are forwarded to the correct egress port. A DUT may have a different Queueput value for each configured Classification.

XOFF Frame: A Priority-based flow control pause frame that instructs the DUT to pause one or more VLAN priorities.

XON Frame: A Priority-based flow control pause frame that instructs the DUT to resume transmission on one or more VLAN priorities.

4. General Considerations

4.1. Classifications

Data Center Bridging devices SHOULD be tested with multiple Classifications. Testing with a single Classification provides no means to test and measure a device's ability to differentiate forwarding behavior for different traffic classes.

4.2. Congestion

For devices capable of forwarding traffic at line rate, explicit congestion MUST be created via the test tool to benchmark queue

performance. Possible methods for accomplishing this on a DUT with n ports include, but are not limited to:

1. Test full-mesh traffic patterns on $(n-1)$ ports while using 1 port as a multicast transmitter with $(n-1)$ multicast receivers.
2. Test full-mesh traffic patterns on $(n-1)$ ports while generating partially meshed traffic between 1 and $(n-1)$ ports.
3. Use partially meshed traffic patterns with x ports transmitting to y ports where $x > y$ and $x + y = n$.

4.3. Test Traffic

The lock-step traffic pattern, as described in section 5.1.3 of [RFC2889], is specifically NOT required for DCB testing for two reasons:

1. Such patterns are not meaningful for high speed Ethernet devices due to the transmission clock variance allowed by the IEEE 802.3 Ethernet specification.
2. Flow control mechanisms would quickly break such patterns when activated.

4.4. Tester Capabilities

4.4.1. Frame Formats

This testing document does not mandate the use of any particular frame format for testing. Any frame that can be legally forwarded by the DUT/SUT MAY be used provided that the test instrument can make the following distinctions for each frame:

1. The test tool MUST be able to distinguish test frames from non-test frames.
2. The test tool MUST be able to determine whether each test frame is forwarded to the correct egress port.
3. The test tool MUST be able to determine whether each received frame conforms or does not conform to the Conformance Vector of the frame's Classification Group, if applicable.

4.4.2. Pause Response Time

To accurately measure the performance of a Priority-based Flow Control capable DUT, the test tool MUST be able to respond to PFC pause frames. Additionally, the test tool MUST respond to all received pause frames in the time period specified in the IEEE 802.1Qbb specification.

5. Test Setup

This document extends the general test setup described in section 3 of [RFC2889] and section 6 of [RFC2544] to the benchmarking of Data Center Ethernet switching devices. [RFC2889] and [RFC2544] describe benchmarking methodologies for networking devices that intentionally drop frames at their performance limits. In DCB networks, the DUT will transmit PFC Pause frames as a Congestion Management method to throttle network endpoints, thus minimizing the probability of frame loss in the network.

5.1. Test Traffic

5.1.1. Traffic Classification

Since DCB devices are expected to support multiple traffic Classifications, it is RECOMMENDED to benchmark DCB devices with multiple Classification Groups.

5.1.2. Trial Duration

The RECOMMENDED trial duration is 300 seconds. However other durations MAY be used. Additionally, a running trial MAY be aborted once the test tool determines that the currently running trial has failed, e.g. QoS bounds exceeded, packet loss detected on a lossless queue, etc.

5.1.3. Frame Measurements

Packet Conformance MUST be determined for all test frames on a per frame basis. The method specified for measuring Latency in [RFC2544], e.g. measuring the latency of a single test frame in a traffic flow, is unsuitable for DCB benchmarking.

5.1.3.1. Forwarding Delay and Latency

Multiple methods exist for measuring the time it takes a test frame to be forwarded by a DUT. However, both of the methods discussed in [RFC1242] are unsuitable for testing DCB devices, as many DCB devices

alternate between both "store and forward" and "bit forwarding" behavior depending upon their queue congestion. Hence, the only RECOMMENDED method for measuring the time it takes a DUT to forward a test frame is "Forwarding Delay" as described in [RFC4689].

5.1.4. Frame Sizes

5.1.4.1. Ethernet

The recommended frame sizes for Ethernet testing are 64, 128, 256, 512, 1024, 1280, 1518, 4096, 8192, and 9216 as per [RFC5180]. Note that these frame sizes include the Ethernet CRC and VLAN header.

5.1.4.1.1. Fiber Channel over Ethernet

FCoE test traffic introduces a number of frame size constraints that make the default frame sizes specified in [RFC5180] unusable:

1. FCoE frames contain an encapsulated Fiber Channel frame. Due to the method of encapsulation used, all FCoE frames MUST be a multiple of 4 bytes. See [RFC3643].
2. Test tools may need to include a test payload in addition to the encapsulated Fiber Channel frame to meet the requirements specified in Section 4.4.1.
3. The maximum supported frame size for FCoE is 2176 bytes.

Due to these constraints, the recommended frame sizes for FCoE testing are 128, 256, 512, 1024, 1280, 1520, 2176, and the smallest FCoE frame size supported by the test tool. Note that these frame sizes include both the Ethernet CRC and VLAN header.

5.1.5. Burst Sizes

As per [RFC2285], the burst size specifies the number of test frames in a burst. To simulate bursty traffic, the test tool MAY send a burst of test traffic with the minimum, legal Inter-Frame Gap (IFG) between frames in the burst followed by a larger Inter-Burst Gap (IBG) between sequential bursts. Note that burst sizes are only applicable to test traffic when the Offered Load of the test ports is less than the Maximum Offered Load (MOL) of those ports. Additionally, a burst size of 1 specifies a constant load, e.g. non-bursty traffic.

6. Benchmarking Tests

6.1. Pause Response Time

6.1.1. Objective

To determine the amount of time required for the DUT to respond to priority-based flow control pause frames.

6.1.2. Setup Parameters

The following parameters MUST be defined. Each variable is configured with the following considerations.

Each Classification Group MUST be listed. For each classification group, the following parameters MUST be specified:

Codepoint - For DCB tests, the codepoint is the VLAN priority.

Frame Size - The frame size includes both the CRC and VLAN header. See Section 5.1.4 for recommended frame sizes.

Burst Size - The burst size specifies the number of frames transmitted with the minimum legal IFG before pausing. See Section 5.1.5.

Intended Vector - The intended vector SHOULD specify the intended rate of test traffic specified as a percentage of port load.

Traffic Pattern - The traffic distribution and traffic orientation used for this Classification.

Conformance Vector - The conformance vector is optional, but MUST be defined if used.

Priority-based Flow Control - PFC mechanisms MUST be enabled.

Background Traffic - Background traffic MAY be present.

PFC Pause Parameters:

Queue(s) - A list of one or more VLAN priorities the test tool should attempt to pause.

Pause Value - The quanta value to use in the XOFF frame(s).

XON Delay - The amount of time to pause the DUT before sending a XON frame. Note that if the XON Delay is larger than the Pause Value, the test tool MUST send multiple XOFF frames to ensure that the DUT remains paused until the XON frame is transmitted.

6.1.3. Procedure

The test tool SHOULD generate test traffic for at least 30 seconds before sending any XOFF frame in order for the DUT to reach a steady-state forwarding condition. The test tool then transmits one or more XOFF frames on one or more ports. Each XOFF frame SHOULD instruct the DUT to pause one or more of the Classification Groups currently being forwarded by the DUT. The test tool MAY optionally send a XON frame to instruct the DUT to resume transmission.

6.1.4. Measurements

The following measurements MUST be reported for each test port and codepoint involved in the test.

Offered Load - the Offered Load from the DUT in N-octet frames per second or bits per second. Note: The Offered Load from the DUT may be insufficient to accurately measure the DUT's Pause Response Time. This condition SHOULD be noted in the results.

The total number of PFC frames transmitted to the DUT by the test tool.

The following values SHOULD be reported in either quanta OR seconds:

Pause Response Time - The time between the transmit time of the last bit of the pause frame and the receive time of the first bit of the last codepoint matching test frame forwarded by the DUT before the DUT is observed to pause the intended queue.

Intended Pause Time - The total time the test tool instructed the DUT to pause.

Observed Pause Time - The actual time the DUT was observed to pause.

XON Response Time - The time between the transmit time of the last bit of the XON frame and the receive time of the first bit of the first unpaused test packet from the DUT.

6.1.5. Reporting Format

TBD

6.2. Queueput

6.2.1. Objective

To determine the Queueput for one or more Traffic Classifications of a DUT using priority flow control.

6.2.2. Setup Parameters

The following parameters MUST be defined. Each variable is configured with the following considerations.

Each Classification Group MUST be listed. For each classification group, the following parameters MUST be specified:

Codepoint - For DCB tests, the codepoint is the VLAN priority.

Frame Size - The frame size includes both the CRC and VLAN header. See Section 5.1.4 for recommended frame sizes.

Burst Size - The burst size specifies the number of frames transmitted with the minimum legal IFG before pausing. See Section 5.1.5.

Intended Vector - The intended vector SHOULD specify the intended rate of test traffic specified as a percentage of port load.

Traffic Pattern - The traffic distribution and traffic orientation used for this Classification.

Conformance Vector - The conformance vector is optional, but MUST be defined if used.

Priority-based Flow Control - PFC mechanisms MUST be enabled.

Background Traffic - Background traffic MAY be present.

6.2.3. Procedure

A search algorithm is used to determine the Queueput for each Classification Group. If Queue Congestion is detected for a Classification Group during a trial, then the Intended Vector for the Classification Group MUST be reduced for the subsequent trial. If a

Conformance Vector is specified for the test and Non-conformant frames are received during a trial, then the Intended Vector SHOULD be reduced for the subsequent trial. The algorithm MUST adjust the Intended Vector for each Classification Group. The search algorithms for each Classification Group MAY be run in parallel. The test continues until all Classification Groups in the test have converged on a discrete Queueput value.

6.2.4. Measurements

The Queueput for each Classification MUST be reported in either N-octet frames per second or bits per second.

If a Conformance Vector is specified for a Classification Group, any Non-conformant frames MUST be reported.

The number of PFC pause frames transmitted by the DUT for each code-point in the Codepoint Set MUST be reported for each test port.

The total pause time observed by the tester for each code-point in the Codepoint Set MUST be reported for each test port.

Any frame loss observed for test traffic using PFC enabled codepoints MUST be reported. Any frame loss observed for test traffic using non-PFC enabled codepoints on uncongested egress ports SHOULD be reported, as that indicates the DUT is performing Head of Line Blocking (HOLB).

6.2.5. Reporting Format

TBD

6.3. Maximum Forwarding Rate

6.3.1. Objective

To determine the maximum forwarding rate of one or more PFC queues on a PFC capable DUT.

6.3.2. Setup Parameters

Maximum Forwarding Rate is conceptually similar to the measurement in [RFC2285] but works on a per-Classification basis in a DCB context. The following parameters MUST be defined. Each variable is configured with the following considerations.

Each Classification Group MUST be listed. For each classification group, the following parameters MUST be specified:

Codepoint - For DCB tests, the codepoint is the VLAN priority.

Frame Size - The frame size includes both the CRC and VLAN header. See Section 5.1.4 for recommended frame sizes.

Burst Size - The burst size specifies the number of frames transmitted with the minimum legal IFG before pausing. See Section 5.1.5.

Intended Vector - The intended vector includes the intended rate of test traffic specified as a percentage of port load.

Traffic Pattern - The traffic distribution and traffic orientation used for this Classification.

Conformance Vector - The conformance vector is optional, but MUST be defined if used.

Priority-based Flow Control - PFC mechanisms SHOULD be disabled.

Background Traffic - Background traffic MAY be present.

6.3.3. Procedure

The tester should iterate across all configured permutations of frame size, burst size, and Intended Vector for all Classification Groups.

6.3.4. Measurements

The forwarding rate of each Classification Group MUST be reported as the number of N-octet test frames per second the DUT correctly forwards to the proper egress port.

The maximum forwarding rate for each Classification Group MUST be reported as the highest recorded forwarding rate from the set of all iterations.

Both the Intended and Offered Vector of each Classification Group MUST be reported.

If a Conformance Vector is specified for a Classification Group, any Non-conformant frames MUST be reported.

The number of PFC pause frames transmitted by the DUT for each code-point in the Codepoint Set MUST be reported.

The total pause time observed by the tester for each code-point in the Codepoint Set MUST be reported.

6.3.5. Reporting Format

TBD

6.4. Back-off

6.4.1. Objective

To determine the delta between the maximum forwarding rate of a DUT and the point where the DUT ceases to use PFC to manage priority queues.

6.4.2. Setup Parameters

The following parameters MUST be defined. Each variable is configured with the following considerations.

Each Classification Group MUST be listed. For each classification group, the following parameters MUST be specified:

Codepoint - For DCB tests, the codepoint is the VLAN priority.

Frame Size - The frame size includes both the CRC and VLAN header. See Section 5.1.4 for recommended frame sizes.

Burst Size - The burst size specifies the number of frames transmitted with the minimum legal IFG before pausing. See Section 5.1.5.

Intended Vector - The intended vector includes the intended rate of test traffic specified as a percentage of port load.

Traffic Pattern - The traffic distribution and traffic orientation used for this Classification.

Conformance Vector - The conformance vector is optional, but MUST be defined if used.

Priority-based Flow Control - PFC mechanisms MUST be enabled.

Backoff method - The recommended backoff method is to reduce the aggregate traffic load by a fixed amount while still maintaining a fixed load ratio between all Classification Groups.

6.4.3. Procedure

The initial trial SHOULD begin with an Intended Load equal to or greater than the Maximum Forwarding Rate of the DUT/SUT. For each subsequent trial, the aggregate load is reduced until the DUT is observed to complete a trial without activating any Congestion Management methods.

6.4.4. Measurements

The Intended and Offered Vector for each Classification Group MUST be reported.

If a Conformance Vector is specified for a Classification Group, any Non-conformant frames MUST be reported.

The number of PFC pause frames transmitted by the DUT for each code-point in the Codepoint Set MUST be reported.

The total pause time observed by the tester for each code-point in the Codepoint Set MUST be reported.

Any frame loss observed for test traffic using PFC enabled codepoints MUST be reported. Any frame loss observed for test traffic using non-PFC enabled codepoints on uncongested egress ports SHOULD be reported, as that indicates the DUT is performing Head of Line Blocking (HOLB).

6.4.5. Reporting Format

TBD

6.5. Back-to-Back

6.5.1. Objective

To determine the maximum duration a DUT can forward test traffic with minimum Inter-Frame Gap on one or more PFC queues without using Congestion Management.

6.5.2. Setup Parameters

The following parameters MUST be defined. Each variable is configured with the following considerations

Each Classification Group MUST be listed. For each classification group, the following parameters MUST be specified:

Codepoint - For DCB tests, the codepoint is the VLAN priority.

Frame Size - The frame size includes both the CRC and VLAN header. See Section 5.1.4 for recommended frame sizes.

Intended Vector - The intended vector includes the intended rate of test traffic specified as a percentage of port load.

Traffic Pattern - The traffic distribution and traffic orientation used for this Classification.

Conformance Vector - The conformance vector is optional, but MUST be defined if used.

Priority-based Flow Control - PFC mechanisms MUST be enabled.

The sum of all Intended Vectors on a transmitting port SHOULD equal the Maximum Offered Load (MOL) of that port.

6.5.3. Procedure

A search algorithm is used to determine the maximum duration in seconds for which the configured Classification Profile can be forwarded by the DUT without active Congestion Management. If Congestion Management is detected during an iteration, then the duration MUST be reduced for the next iteration.

6.5.4. Measurements

The Intended and Offered Vector for each Classification Group MUST be reported.

If a Conformance Vector is specified for a Classification Group, any Non-conformant frames MUST be reported.

The number of PFC pause frames transmitted by the DUT for each code-point in the Codepoint Set MUST be reported.

The total pause time observed by the tester for each code-point in the Codepoint Set MUST be reported.

Any frame loss observed for test traffic using PFC enabled codepoints MUST be reported. Any frame loss observed for test traffic using non-PFC enabled codepoints on uncongested egress ports SHOULD be reported, as that indicates the DUT is performing Head of Line Blocking (HOLB).

6.5.5. Reporting Format

TBD

7. Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the constraints specified in the sections above.

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT/SUT.

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

8. IANA Considerations

Testers SHOULD use network addresses assigned by IANA for the purpose of testing networks.

9. Normative References

- [RFC1242] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2285] Mandeville, R., "Benchmarking Terminology for LAN Switching Devices", RFC 2285, February 1998.
- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.
- [RFC2889] Mandeville, R. and J. Perser, "Benchmarking Methodology for LAN Switching Devices", RFC 2889, August 2000.

- [RFC3643] Weber, R., Rajagopal, M., Travostino, F., O'Donnell, M., Monia, C., and M. Merhar, "Fibre Channel (FC) Frame Encapsulation", RFC 3643, December 2003.
- [RFC4689] Poretsky, S., Perser, J., Erramilli, S., and S. Khurana, "Terminology for Benchmarking Network-layer Traffic Control Mechanisms", RFC 4689, October 2006.
- [RFC5180] Popoviciu, C., Hamza, A., Van de Velde, G., and D. Dugatkin, "IPv6 Benchmarking Methodology for Network Interconnect Devices", RFC 5180, May 2008.

Appendix A. Acknowledgements

Authors' Addresses

Timmons C. Player
Spirent Communications

Email: timmons.player@spirent.com

David Newman
Network Test

Email: dnewman@networktest.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: April 22, 2012

I. Varlashkin
Easynet Global Services
R. Papneja
Huawei Technologies (USA)
B. Parise
Cisco
T. Van Unen
Ixia
October 20, 2011

Convergence benchmarking on contemporary routers
draft-varlashkin-router-conv-bench-00

Abstract

This document specifies methodology for benchmarking convergence of routers without making assumptions about relation and dependencies between data- and control-planes. Provided methodology is primary intended for testing routers running BGP and some form of link-state IGP with or without MPLS. It may also be applicable for environments using MPLS-TE or GRE, however they're beyond scope of this document and such application is left for further study.

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 <http://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 April 22, 2012.

Copyright Notice

Copyright (c) 2011 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

(<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1. Introduction	4
2. Test topology	5
3. TEST PARAMETERS	6
3.1. Packing ratios	7
3.2. Test traffic	7
3.3. IGP metrics	7
3.4. Internal routers matrix	7
3.5. Number of next-hops	8
3.6. 'e' - Failure and Restoration start entropy	8
4. TEST PROCEDURES	8
4.1. Initialisation time	8
4.2. Generic data-plane failure test	9
4.3. Generic test procedure for	9
5. Failure and restoration scenarios	10
5.1. Loss of Signal on the link attached to DUT	10
5.2. Link failure without LoS	10
5.3. Non-direct link failure	11
5.4. Best route withdrawal	11
5.5. iBGP next-hop failure	12
6. Test report	12
7. Link bundling and Equal Cost Multi-Path	13
8. Graceful Restart and Non-Stop Forwarding	13
9. Security considerations	13
10. IANA Considerations	14
11. Acknowledgments	14
12. Normative References	14
Authors' Addresses	14

1. Introduction

Ability of the network to restore traffic flow when primary path fails has always been important subject for network engineers, researchers and equipment manufacturers. Time to recover from a link or node failure has often been linked to routing protocols convergence; and benchmarking of a routing protocol convergence has often been considered sufficient for quantifying recovery performance. As long as routers could obtain new best path only after relevant routing protocols perform their calculations such methodology was reasonable. However continuous improvements in hardware and software result in more and more routers being able to restore traffic flow even before routing protocols converge. Methodology described in this document takes such fact into account.

When a failure occurs on the network a router needs to:

1. select new best path so that the packets, which already arrived to the router, can be forwarded
2. let other routers know about new network state so they can find new best path from their perspective

How fast a router can perform these two functions characterise router's performance with regards to convergence. Note that in general case each of these characteristics may or may not be related to the other. For example, some platform may need to perform calculations to find new best path and only then update local FIB and send relevant protocol updates to other routers, another platform can update local FIB without waiting for calculations to complete but still needs to wait for calculations before sending routing protocol updates, third platform can use different optimisation for both FIB changes and routing protocol updates without waiting for completion of the calculations. Other variations are also possible. This document makes no assumption about whether local FIB changes and routing protocol updates dependencies on each other or on routing protocol calculations.

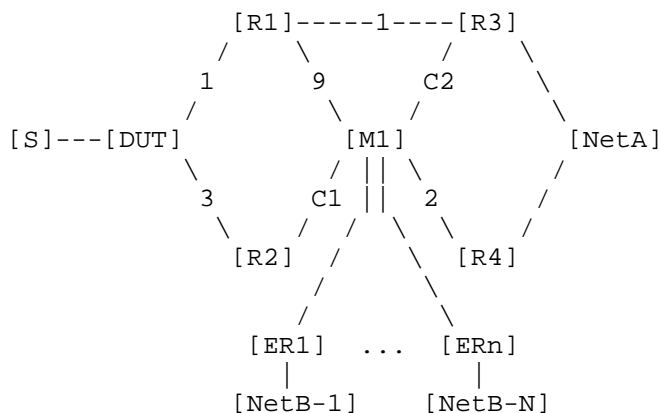
Since it is not known whether local FIB is updated before or after routing protocol calculations, forwarding-plane method is proposed to benchmark local convergence. And because it is not known whether routing protocol updates are linked to FIB modification or not the control-plane approach is used to benchmark how fast updates are propagated. However both characteristics are benchmarked using very similar test topologies and procedures. Also, an attempt is made to to minimise dependency on performance on non-DUT elements involved in the tests.

At the time of writing of this document it is not known whether existing network testers and protocol emulators are able to execute described tests out of the box. Nevertheless the authors believe that required functionality can be added with reasonable effort. Alternatively the tests can be performed with help of physical routers to create necessary test topology, which may have impact on time required to perform the test but expected to provide same degree of the test results accuracy. This also means that tests performed using a protocol simulator can be repeated using physical routers and results expected to be comparable.

This document complements draft-papneja-bgp-basic-dp-convergence.

2. Test topology

Unless specified otherwise all tests use same basic test topology outlined below:



S is source of test traffic for data-plane tests, while for control-plane tests S is an emulated or physical router with packet capturing (sniffing) capability.

Unidirectional test traffic goes from Source to NetA.

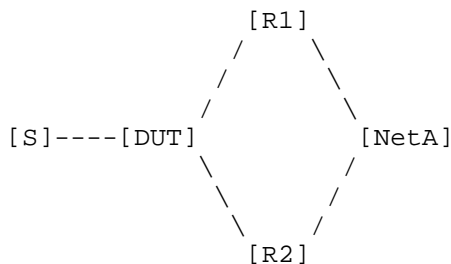
IGP between DUT and R1-R4; BGP between DUT and R3, R4; no BGP between R3 and R4 (important). If tunnelling (e.g. MPLS or GRE) is used then R1 and R2 do not need to run BGP, otherwise they MUST run BGP. Source has static default to DUT; R3 and R4 have static to NetA. NetA is in BGP but not in IGP. M1 is K*M matrix of internal routers. Metrics C1 is used to control whether R2 is LFA for DUT to NetA. Metric C2 is used to control whether R3 or R4 are best exit towards NetA. All other metrics are fixed for all tests and MUST be set to

exact values provided in the above diagram. IGP metrics from M1 to ER1 throughout ERn can be set arbitrarily, their exact values are irrelevant to this test as long as they're valid for given IGP.

Routers ER1 throughout ERn together with prefixes NetB-1 throughout NetB-N are presented to create realistic environment but not used directly in measurements. NetB-1 throughout NetB-N are distinct single-prefix sets.

Traffic restoration depends on ability of R2 and M1 to forward traffic after failure. To eliminate this dependency R2 is set to always forward traffic to R3 and NetA via M1 which in turn always forwards traffic directly via R3 or R2 depending on the test. One possibility to achieve this is to use static routes. Another alternative is to use different IGP between R2 and R3 from the one used by DUT and make routes learned via this IGP preferred on R2. E.g. DUT uses OSPF, then in addition to it R2&R3 also run ISIS and prefer ISIS routes over OSPF ones. A protocol simulator can have internal mechanism to provide required behaviour. There are no other dependencies on non-DUT devices in this tests.

For evaluating eBGP performance following topology is used:



Test topology for eBGP

In "Link failure without LoS" test direct cable between DUT and R1 is replaced with connection over an L2 switch as follow:

[DUT]---[SW1]---[R1]

3. TEST PARAMETERS

3.1. Packing ratios

Routes with different prefixes but same attributes can potentially be packed into single update message. Since both number of update messages and number of prefixes per update can affect convergence time, the tests SHOULD be performed with various prefix packing ratios. This document does not specify values of individual BGP attributes used to control packing ratio.

3.2. Test traffic

Traffic is sent from single source address located at the Source port of the tester to one address in each prefix in NetA set. Packets are sent at rate 1000 per second, which provides 1ms resolution of the convergence time as measured by tests in this document. All packets SHOULD be 64 bytes at IP layer, that is IP header plus IP payload.

3.3. IGP metrics

Basic test topology specifies fixed IGP metrics for some links. These metrics SHOULD be used verbatim. There are also two variable metrics - C1 and C2 - intended for controlling whether R2 is Loop-Free-Alternate (LFA) for DUT towards NetA, and whether R3 remains best exit towards NetA after path failure between DUT and R3. Following values SHOULD be used for C1 and C2 depending on required behaviour:

R2 is LFA?	R3 best?	C1	C2
yes	yes	1	1
yes	no	1	3
no	yes	5	1
no	no	5	3

3.4. Internal routers matrix

Basic test topology has N*K grid of internal routers denoted as M1. When N>1 or K>1 the cost of all links within grid MUST be set to 1 (one). This matrix is intended for controlling topology size, which has affect on particularly SPF run-time.

If traffic is forwarded using a tunneling mechanism, such as MPLS or GRE, the internal routers only need to have reachability information about tunnel end-points. However if traditional hop-by-hop forwarding is used, then internal routers MUST have routes to each and every prefix within NetA set.

This document does not specify how internal routers should obtain necessary reachability information. The only requirement is that after primary DUT-NetA path failure internal routers are able to forward traffic to NetA instantly. Using values of IGP metrics as described earlier addresses this requirement. Also, protocol simulator may have built-in mechanism to achieve desired behaviour.

3.5. Number of next-hops

Basic test topology has set of N edge routers ER1 throughout ERn, each advertising unique prefix. Some BGP implementations may exhibit different performance depending on number of next-hops for which IGP cost has changed after failure. By varying overall number of next-hops such dependency can be detected.

Note that prefixes NetB-1 throughout NetB-n are not used as destinations for test traffic, they're only present for creating "background environment".

3.6. 'e' - Failure and Restoration start entropy

Tests described in this document use fixed time T2 and variable offset 'e' as starting point for simulating failure or restoration event.

Fixing time T2 is necessary as reference point to which variable offset e is added for each iteration of the test. Introduction of such variable offset allows better analysis of the test results. For example, DUT may run FIB changes at certain intervals. If failure introduced close to the end of such interval, shorter outage will be observed, and if introduced close to the beginning of such interval longer outage will be observed. Running test multiple times each time using different offset will help to profile DUT better.

Test report must contain value of T2 (same for all iterations) and values of e for each iterations. This document recommends to use $T2=T1+8s$ and e from 0 to 1s in 0.01s (10ms) increments.

4. TEST PROCEDURES

This section provides generic steps that are used in all tests.

4.1. Initialisation time

The objective of this test is to measure time that must elapse between starting protocols and ability of the test topology to forward traffic. This test is not intended to reflect DUT

performance but used only as a way to find time T_1 that is used in all subsequent tests.

To execute test perform following steps:

1. Configure DUT and protocol simulator (or auxiliary nodes)
2. At T_0 start traffic and then immediately start routing protocols
3. When traffic starts arriving Sink Port 1 stop test.

The time of arrival of the first packet is T_1 .

4.2. Generic data-plane failure test

The purpose of failure test is to measure time required by DUT to resume traffic flow after best path to destination fails. Following steps are common for all failure tests:

1. Start protocols and mark time as T_0
2. At time T_1 start traffic to each prefix in set NetA
3. At $T_2 + e$ simulate failure or restoration event (see Section 5)
4. From $T_2 + e$ until T_3 packets do not arrive to NetA
5. After packets are seen again at NetA (T_3) wait until time T_4
6. Stop traffic
7. Measure total number of lost packets and calculate outage knowing packet-per-second

4.3. Generic test procedure for

1. At T_0 bring up all interfaces and protocols, and start capturing BGP packets at RS1
2. At $T_1 + e$ simulate failure/restoration event (see Section 5)
3. At $T_2 - d_1$ first UPDATE message is sent by DUT and at T_2 it will be observed at RS1
4. At $T_3 - d_2$ last UPDATE message is sent by DUT and at T_3 it will be observed at RS1

d_1 and d_2 represent serialisation and propagation delay and can be

disregarded unless DUT-RS1 link has large delay. With this in mind, T2-(T1+e) and T3-(T1+e) represent convergence time for the first and last prefix respectively.

5. Failure and restoration scenarios

This section defines set of various failure and restoration scenarios used in step 3 of the generic test procedures described in previous section. Unless otherwise specified all scenarios are applicable to both data- and control-plane test procedures.

5.1. Loss of Signal on the link attached to DUT

This scenario simulates situation where link attached to DUT fails and Loss of Signal (LoS) can be observed by DUT. In other words link fails and results in interface on the DUT going down.

To simulate LoS failure at the time defined by the test procedure shut down R1 side of the link to DUT.

To simulate LoS restoration at the time defined by the test procedure re-activate R1 side of the link to DUT.

5.2. Link failure without LoS

This scenario simulates situation where link between DUT and adjacent node fails but DUT does not observe LoS. In practice such failure can occur when, for example, link between DUT and adjacent node is implemented via carrier equipment that does not shut link down when remote side of the link fails.

DUT can use various methods to detect such failures, including but not limited to protocol HELLO or Keep-alive packets, BFD, OAM. This document does not restrict methods which DUT can use, but requires use of particular method to be recorded in the test report.

Basic network topology is modified for the purpose of this test only as follow: rather than using direct cabling between DUT and R1 the link is implemented via intermediate L2 switch that supports concept of VLAN's. Initially switch ports connected to DUT and R1 are placed into the same VLAN (same L2 broadcast domain).

To simulate failure at the time defined by the test procedure move switch port connected to R1 to a VLAN different from the one used for switch port connected to DUT.

To simulate restoration at the time defined by the test procedure

move switch port connected to R1 back to the same VLAN as the one used for switch port connected to DUT.

5.3. Non-direct link failure

This scenario simulates situation where a link not directly connected to DUT but located on the primary path to destination fails. Unmodified basic network topology is used.

Depending on technologies used in the setup different failure detection techniques can be employed by DUT. This document assumes that DUT relies exclusively on IGP information to learn about failure and that nodes adjacent to the failed link flood this information within D seconds since the event. If required exact value of D can be obtained through simple additional test, but in this document D is assumed to be 0 (zero).

It is possible, though undesirable, that some traffic and protocol simulators may continue accepting packets coming through the port that leads to simulated failed link. It is essential to assert such behaviour prior to the tests and if confirmed, exclude packets received after failure from calculations in step 7 of the test.

Failure event is triggered by simulating shutdown of R3 side of the link to R1 at the time defined by the test procedure. R1 MUST send IGP update (depending on which protocol is used) to DUT within D seconds.

Restoration event is triggered by simulating recovery of R3 side of the link to R1 at the time defined by the test procedure. R1 MUST send IGP update (depending on which protocol is used) to DUT within D seconds.

5.4. Best route withdrawal

This scenario simulates situation where best AS exit path to a destination is no longer valid and ASBR sends BGP UPDATE to its iBGP peers. Unmodified basic network topology is used.

Disconnecting R3 from NetA implies that R3 will send BGP WITHDRAW for this prefixes in its update to DUT. It is possible, though undesirable, that some protocol simulator and traffic generators will still count packets received at sink port 1 even after prefixes were withdrawn. To correctly execute this test it's mandatory that traffic received at sink port 1 after withdrawing prefixes is ignored and not counted as delivered. If traffic generator is not able to assure such functionality (should be asserted prior to the test), then packets received at the sink port 1 MUST be excluded from

calculation in step 7 of the test.

Failure event is triggered by simulating failure of the link between R3 and NetA and immediate withdrawal of all corresponding prefixes by R3.

Restoration event is triggered by simulating recovery of the link between R3 and NetA and immediate BGP UPDATE for all corresponding prefixes by R3.

5.5. iBGP next-hop failure

This scenario simulates situation where ASBR used as best exit to a destination unexpectedly fails both at control and forwarding plane. Both R1 and a router within M1 connected to R3 MUST send appropriate IGP update message to the rest of the network within D seconds. To detect failure DUT MAY rely on IGP information provided by rest of the network or it MAY employ additional techniques. This document does not restrict what detection mechanism should DUT use but requires that particular mechanism is recorded in the test report.

Failure event is triggered by simulating removal of R3 from the test topology at the time defined by the test procedure, followed by IGP update as described in previous paragraph.

Recovery event is triggered by re-introducing R3 into the test topology, followed by IGP update as described in first paragraph of this section and immediate re-activation of BGP session between R3 and DUT. Note that recovery time calculated by this method depends on DUT performance in respect to bringing up new BGP session. This is intentional. Control plane convergence benchmarking can be performed separately by a method that is outside of the scope of this document and two results can be correlated netto data-plane convergence value should that be necessary.

6. Test report

TODO: Report format is to be discussed.

Test report MUST contain following data for each test:

1. T1 and 'e'
2. Number of prefixes NetA and NetB
3. Size of M1 (recored as N*K)

4. Traffic rate, in packets per second, and packet size at IP layer in octets
5. Number of lost packets during failure, and number of lost packets during restoration

7. Link bundling and Equal Cost Multi-Path

Scenarios where DUT can balance traffic to NetA across multiple best paths is explicitly excluded from scope of this document. There are two reasons.

First, two different DUT may choose different path (out of all equal) to forward given packet, which makes it unreasonably difficult to define generic traffic that would produce comparable results when testing different platforms.

Second, mechanisms used to handle failures in ECMP (but not necessarily in link-bundling) environment are similar to those handling single-path failures. Therefore it's expected that convergence in ECMP scenario will be of the same order as in single-path scenario.

8. Graceful Restart and Non-Stop Forwarding

While Graceful Restart and Non-Stop Forwarding mechanisms are related to DUT ability to forward traffic under certain failure conditions, the test covering DUT own ability to restore or preserve traffic flow already covered in RFC6201.

9. Security considerations

The tests described in this document intended to be performed in isolated lab environment, which inherently has no security implication on the live network of the organisation or Internet as whole.

Authors foresee that some people or organisations might be interested to benchmark performance of the live networks. The tests described in this document are disruptive by their nature and will have impact at least on the network where they're executed, and depending on the role of that network effect can extend to other parts of the Internet. Such tests MUST NOT be attempted in live environment without careful consideration.

The fact of publishing this document does not increase potential negative consequences if tests are executed in live environment because information provided here is mere recording of widely known and used techniques.

10. IANA Considerations

None.

11. Acknowledgments

Authors would like to thank Gregory Cauchie, Rob Shakir, David Freedman, Anton Elita, Saku Ytti, Andrew Yourtchenko, for their valuable contribution and peer-review of this work.

12. Normative References

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

Authors' Addresses

Ilya Varlashkin
Easynet Global Services

Email: ilya.varlashkin@easynet.com

Rajiv Papneja
Huawei Technologies (USA)

Email: rajiv.papneja@huawei.com

Bhavani Parise
Cisco

Email: bhavani@cisco.com

Tara Van Unen
Ixia

Email: TVanUnen@ixiacom.com

