Benchmarking Methodology for IPv6 Transition Technologies

draft-ietf-bmwg-ipv6-tran-tech-benchmarking-00

Marius Georgescu

Nara Institute of Science and Technology
Internet Engineering Laboratory

3 Nov. 2015

Draft Motivation: IPv6 transition

- IPv6 is not backwards compatible
- The Internet will undergo a period through which both protocols will coexist
- Currently only 5% of worldwide Internet users have IPv6 connectivity [1]

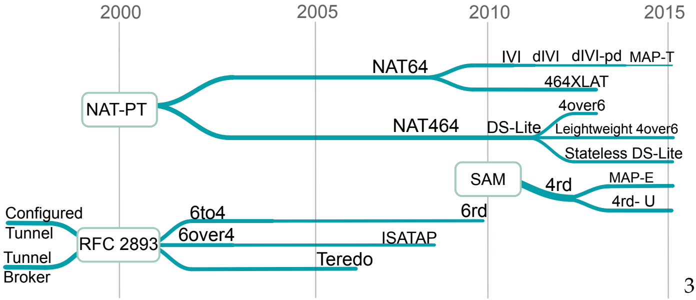
[2]

[1] APNIC. *IPv6 measurements for The World*. Asia-Pacific Network Information Centre, Oct. 2015. URL: http://labs.apnic.net/ipv6-measurement/Regions/.

[2] **Original drawing by Andrew Bell @ www.creaturesinmyhead.com** .

IPv6 transition technologies evolution

- What benchmarks to use?
  - For Dual Stack RFC2544 or RFC5180 are enough
  - How about translation/encapsulation technologies?



[3]

[3] **inspired by the APNIC35 presentation "The evolution of IPv6 transition technologies" by Jouni Korhonen**.

Draft overview

- This draft provides complementary guidelines to RFC2544[4] and RFC5180[5] for evaluating the performance of IPv6 transition technologies
  - generic classification on IPv6 transition technologies → associated test setups
  - calculation formula for the maximum frame rate according to the *frame size overhead*
- Includes a tentative metric for benchmarking scalability
  - scalability as *performance degradation* under the stress of *multiple network flows*
- Proposes supplementary benchmarking tests for *stateful* IPv6 transition technologies in accordance with RFC3511[6]
- Proposes supplementary benchmarking tests for *DNS resolution performance*
  - contributed by Prof. Gábor Lencse [RG profile link]

---

[4] S. Bradner and J. McQuaid. *Benchmarking Methodology for Network Interconnect Devices*. United States, 1999.

[5] A. Hamza C. Popoviciu, G. Van de Velde, and D. Dugatkin. *IPv6 Benchmarking Methodology for Network Interconnect Devices*. RFC 5180. Internet Engineering Task Force, 2008.

[6] B. Hickman et al. *Benchmarking Methodology for Firewall Performance*. RFC 3511 (Informational). Internet Engineering Task Force, Apr. 2003. URL: http://www.ietf.org/rfc/rfc3511.txt.

Draft History & Outlook

- Proposed in IETF91 as:
  draft-georgescu-ipv6-transition-tech-benchmarking-00
- Evolved with comments from BMWG through IETF92 and IETF93 to:
  draft-georgescu-bmwg-ipv6-tran-tech-benchmarking-01
- Call for adoption in the BMWG IETF93 meeting confirmed recently on the BMWG ML
  - **Thank you very much for the feedback and support !!!**
- WG draft 00 available in IETF94:
  draft-ietf-bmwg-ipv6-tran-tech-benchmarking-00
  - Detailed change-log for version 00 [link]
- WGLC maybe in IETF96 or IETF97 depending on the progress

Update overview

- Redefined the generic categories of IPv6 transition technologies using IP-specific domains as context → dropped the CE/PE terminology
- Removed SONET as example media
- Recommended a fixed MTU size: *1280 + encaps/transl overhead*
- Updated the text on Extension Headers following the commments
- Clarified the limitations of the dual DUT setup
- Proposed benchmarking tests for DNS resolution performance
  - Considered DNS caching as a testing parameter
- Added rationale for using *Mean* instead of *Median* as summarizing function
- Various smaller editorial changes (detailed changelog [link])

UPDATE: REDEFINED THE GENERIC CATEGORIES

1. **Single Translation**: the production network is assumed to have only two domains, Domain A and the Core domain. The core domain is assumed to be IPvY specific. IPvX packets are translated to IPvY at the edge between Domain A and the Core domain.
2. **Dual-stack**: the core domain devices implement both IP protocols
3. **Encapsulation**: Domains A and B are IPvX specific, while the core domain is IPvY specific. An encapsulation mechanism is used to traverse the core domain.
4. **Double translation**: Domains A and B are IPvX specific, while the core domain is IPvY specific. A translation mechanism is employed for the traversal of the core network.

X,Y $\in$ {4,6}

[7] **following the comments from Fred Baker**.

Update: Recommended "fixed" MTU

Text added to Section 4.1:

In the context of frame size overhead, MTU recommendations are needed in order to avoid frame loss due to MTU mismatch between the virtual encapsulation/translation interfaces and the physical network interface controllers (NICs). To avoid this situation, the larger MTU between the physical NICs and virtual encapsulation/translation interfaces SHOULD be set for all interfaces of the DUT and tester. **To be more specific, the minimum IPv6 MTU size (1280 bytes) plus the encapsulation/translation overhead is the RECOMMENDED value for the physical interfaces as well as virtual ones.**

[8] **following the comments from Masataka Mawatari** .

UPDATE: EXTENSION HEADERS

Text added to Section 4.2:

Note: testing traffic with extension headers might not be possible for the transition technologies which employ translation. Proposed IPvX/IPvY translation algorithms such as IP/ICMP translation [RFC6145] do not support the use of extension headers.

[9] **following the comments from Al Morton and Fred Baker**.

UPDATE: DUAL DUT LIMITATIONS

Text added to Section 3.2:

One of the limitations of the dual DUT setup is the inability to reflect asymmetries in behavior between the DUTs. Considering this, additional performance tests SHOULD be performed using the single DUT setup.
Note: For encapsulation IPv6 transition technologies, in the single DUT setup, in order to test the decapsulation efficiency, the tester SHOULD be able to send IPvX packets encasulated as IPvY.

[10] **following the comments from Scott Bradner**.

Update: DNS64 Resolution Performance

```
1: AAAA query    +--------------------+
   +-----------|                    |<-------------+
   |           |IPv6   tester   IPv4|              |
   |  +------->|                    |----------+   |
   |  |        +--------------------+ 3: empty |   |
   |  | 6: synt'd                       AAAA,  |   |
   |  |   AAAA +--------------------+ 5: valid A|   |
   |  +--------|                    |<---------+   |
   |           |IPv6     DUT    IPv4|              |
   +---------->|      (DNS64)       |--------------+
               +--------------------+ 2: AAAA query, 4: A query
```

Note: The tester plays the role of DNS client as well as authoritative DNS server.

[11] **following the comments from Nalini Elkins**.

UPDATE: DNS64 TRAFFIC SETUP

1. Query for the AAAA record of a domain name (from client to DNS64 server)
2. Query for the AAAA record of the same domain name (from DNS64 server to authoritative DNS server)
3. Empty AAAA record answer (from authoritative DNS server to DNS64 server)
4. Query for the A record of the same domain name (from DNS64 server to authoritative DNS server)
5. Valid A record answer (from authoritative DNS server to DNS64 server)
6. Synthesized AAAA record answer (from DNS64 server to client)

Update: DNS64 Caching

1. Caching[12]:all the DNS queries MUST contain different domain names (or domain names MUST NOT be repeated before the cache of the DUT is exhausted). Then new tests MAY be executed with 10%, 20%, 30% domain names which are repeated (early enough to be still in the cache).
2. Existence of AAAA record:all the DNS queries MUST contain domain names which do not have an AAAA record and have exactly one A record. Then new tests MAY be executed with 10%, 20%, 30%, etc. domain names which have an AAAA record.

[12]**following the comments from Kaname Nishizuka**.

Update: DNS64 Resolution Performance

Text added to Section 8.2:

Objective: To determine DNS64 performance by means of the number of successfully processed DNS requests per second.
Procedure: Send a specific number of DNS queries at a specific rate to the DUT and then count the replies received in time (within a predefined timeout period from the sending time of the corresponding query, having the default value 1 second) from the DUT. If the count of sent queries is equal to the count of received replies, the rate of the queries is raised and the test is rerun. If fewer replies are received than queries were sent, the rate of the queries is reduced and the test is rerun.
Reporting format: The primary result of the DNS64/DNS46 test is the average of the number of processed DNS queries per second measured with the above mentioned "0% + 0% combination". The average SHOULD be complemented with the margin of error to show the stability of the result.

Update: Mean vs. Median

1. Mean is a more inclusive summarizing function
   - Outliers may indicate a malfunction or a bad implemenation in the DUT
2. Mean is less exposed to statistical uncertainty
   - *"...median is less sensitive (more robust) to outliers, but on the other hand has a larger uncertainty in the absence of outliers."*[13]

[13] Robin de Nijs and Thomas Levin Klausen. "On the expected difference between mean and median". In: *Electronic Journal of Applied Statistical Analysis* 6.1 (2013), pp. 110–117.

[14] **following the comments from Kostas Pentikousis**.

Comments not covered yet

- The comment from Fred Baker about Network Performance Degradation: Incremental vs Simultaneous (State Creation vs State Use)
  - Considering 2 separate benchmarking procedures
- Fred's suggestion to use this methodology to benchmark NAT XX as well

Next steps

- DNS Resolution Performance for **DNS46**
- IPv6 transition technologies association table for current proposed technologies as Appendix

⋆ Questions for BMWG:

- Were the comments covered well enough?

Contact


Nara Institute of Science and Technology
NAIST
Marius Georgescu
liviumarius-g@is.naist.jp
www.ipv6net.ro
IPv6
IPv4