Recommendations for using Multiple IP Addresses in Benchmarking Tests

draft-lencse-bmwg-multiple-ip-addresses

Gábor LENCSE lencse@sze.hu (Széchenyi István University) – presenter

Keiichi SHIMA keiichi.shima@g.softbank.co.jp (SoftBank)

IETF 118, BMWG, November 6, 2023.

Outline

- Problem description: why testing with multiple IP addresses is needed?
- Recommended Solution
  - As for IPv4, using the limited IPv4 address range
  - As for IPv6, using the abundant IPv6 address range
  - Question of ranges to be used
- Working code

Problem Description: Conditions

- RFC 2544 has defined a test frame format with fixed IP addresses and fixed port numbers.
- RFC 4814 introduced pseudorandom port numbers, but it kept the usage of a single source and destination IP address pair when a single destination network is used.
- Receive Side Scaling (RSS) supports the receiving of multi million packets per second by distributing the load among CPU cores
  - Depending on implementation, the hash function includes:

    1st type: source IP, destination IP, source port, destination port

    2nd type: source IP, destination IP

Problem Description: Unfairness

- RFC 4814 pseudorandom port numbers + 1st RSS implementation
  - Works perfectly (port numbers ensure entropy)
  - All CPU cores are used, load is distributed approximately evenly
- RFC 4814 pseudorandom port numbers + 2nd RSS implementation
  - Gives poor results (no entropy is ensured as IP addresses are fixed)
  - Thus only two CPU cores are used (one core per direction)
- However, network interconnect devices using the 2nd RSS implementation work perfectly, when they forward Internet traffic (IP addresses ensure entropy)

→ Conditions for the laboratory tests should be improved!

Recommended Solution

- Basic idea: Let us use pseudorandom IP addresses!
  - This is the spirit of RFC 4814 applied to the IP addresses
- Problems to solve:
  - What ranges can be used?
    - There is scarcity in IPv4 addresses reserved for benchmarking
      - 198.18.0.0/15 was reserved for benchmarking
    - There is abundance in IPv6 addresses reserved for benchmarking
      - 2001:2::/48 was reserved for benchmarking
  - What ranges should be used?
    - A trade-off is pointed out

What IPv4 ranges can be used?

- Reserved: 198.18.0.0/15, it is to be cut into two halves:
  - Left side: 198.18.0.0/16 and Right side: 198.19.0.0/16
- RFC 2544 requirement:
  - First, the test suite SHOULD be run with a single source and destination address pair.
    - Typically used: 198.18.0.2/24 and 198.19.0.2/24
  - Then, the tests SHOULD be repeated using 256 different destination networks (chosen randomly)
    - Destination networks denoted by the 16-23 bits of the above network addresses: 198.18.**R**.0/24 and 198.19.**R**.0/24.

    → In this case, only the last 8 bits are available to describe multiple IP addresses

IPv4 Test Setup (multiple destination networks)

- .1 is for the tester; .2 to .254 can be used

```
198.18.0.2/24-198.18.0.254/24      198.19.0.2/24-198.19.0.254/24
           \  +----------------------------------+  /
            \ |                                  | /
+-------------|              Tester              |<------------+
|             |                                  |             |
|             +----------------------------------+             |
|                                                              |
|             +----------------------------------+             |
|             |                                  |             |
+------------>|        DUT: IPv4 router          |-------------+
198.18.0.1/24 |                                  | 198.19.0.1/24
              +----------------------------------+
```

IPv4 Test Setup (single destination network)

- .0.1 is for the tester; .0.2 to .255.254 can be used

```
198.18.0.2/16-198.18.255.254/16  198.19.0.2/16-198.19.255.254/16
           \  +----------------------------------+  /
            \ |                                  | /
+-------------|              Tester              |<------------+
|             |                                  |             |
|             +----------------------------------+             |
|                                                              |
|             +----------------------------------+             |
|             |                                  |             |
+------------>|        DUT: IPv4 router          |-------------+
198.18.0.1/16 |                                  | 198.19.0.1/16
              +----------------------------------+
```

IPv6 Test Setup (in all cases)

- E.g., bits 56-63 can be used for 256 destination networks

```
2001:2::[0000-ffff]:2/64         2001:2:0:8000::[0000-ffff]:2/64
           \  +----------------------------------+  /
            \ |                                  | /
+-------------|              Tester              |<------------+
|             |                                  |             |
|             +----------------------------------+             |
|                                                              |
|             +----------------------------------+             |
|             |                                  |             |
+------------>|        DUT: IPv6 router          |-------------+
            / |                                  | \
           /  +----------------------------------+  \
   2001:2::1/64                              2001:2:0:8000::1/64
```

What ranges should be used?

- On the one hand, the more IP addresses are used, the more entropy is ensured and thus the most even distribution of the load over the processing elements can be expected.
- However, one the other hand, the usage of multiple IP addresses has its costs: multiple Address Resolution Protocol (ARP for IPv4) or Neighbor Discovery Protocol (NDP, for IPv6) table entries are used.
  - Increasing them over a few thousands may have a deteriorating effect on the performance of the DUT.
- More research is needed to give a good recommendation

Working Code

- As a proof of concept, the recommended solution has been implemented in **`siitperf`**.
  - free software under GPL 3 Free Software Free as in Freedom license, available from GitHub <u>://github.com/lencsegabor/siitperf</u>
  - Multiple IPv4 and IPv6 addresses are supported from commit number 165cb7f on September 6, 2023.

We would like to ask for feedback

- Do you agree that the highlighted problem really exist?
- Do you think that the proposed method can solve it?
- Do you have any idea what to change, add, etc.?
- All you comments and suggestions are welcome!