SZÉCHENYI EGYETEM
UNIVERSITY OF GYŐR

IIJ INNOVATION INSTITUTE

Benchmarking Methodology for Stateful NATxy Gateways using RFC 4814 Pseudorandom Port Numbers

draft-lencse-bmwg-benchmaring-stateful

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

Keiichi SHIMA keiichi@iijlab.net (IIJ Innovation Institute)

IETF 112, BMWG, November 8, 2021.

Reminder: Test Setup

- Methodology works with any IP versions
  - To facilitate easy understanding, we use the example of stateful NAT44

```
              +--------------------------------------+
     10.0.0.2 |Initiator                    Responder| 198.19.0.2
+-------------|                Tester                |<------------+
| private IPv4|                         [state table]| public IPv4 |
|             +--------------------------------------+             |
|                                                                  |
|             +--------------------------------------+             |
|    10.0.0.1 |                DUT:                  | 198.19.0.1  |
+------------>|        Sateful NATxy gateway         |-------------+
  private IPv4|     [connection tracking table]      | public IPv4
              +--------------------------------------+
```

Reminder: Measurements in two Phases

- Preliminary test phase
  - It serves two purposes:
    - The connection tracking table of the DUT is filled.
    - The state table of the Responder is filled with valid four tuples.
  - It can be used without the real test phase to measure the maximum connection establishment rate.
- Real test phase
  - It MUST be preceded by a preliminary test phase.
  - The actual measurement procedure (throughput, frame loss rate, latency, PDV, IPDV) is performed as defined in RFC 8219.

Updates since version “-00”

- Version “-01” (August 27)
  - Changes triggered by the comments on the mailing list before IETF 111
- Version “-02” (October 10)
  - Changes triggered by our benchmarking experience with the iptables stateful NAT44 implementation for https://datatracker.ietf.org/doc/html/draft-lencse-v6ops-transition-scalability
    - Complete reworking of Section 4.3
    - Some consequential changes in Section 4.4

Motivation

- Our experience shows that:
  - The processing of a test frame by the sateful DUT requires much more computing power, when it creates a new entry in the connection tracking table, than when it does not create a new entry.
  - The complete depletion of the connection tracking table is even slower than its filling.

Consequence

- There are two extreme situations that we can simply ensure
    1. When all test frames create a new connection
        - Ideal for measuring maximum connection establishment rate
    2. When test frames never create a new connection
        - Ideal for all other tests: throughput, latency, frame loss rate, PDV, etc.
- Due to black box testing, we cannot ensure situations in between
    - E.g. 10% of the test frames create new connections
        - Problem: the content of the connection tracking table of the DUT is not visible for the tester.

Assumptions

- A single source address destination address pair is used for all tests.
  - We make this assumption for simplicity.
  - We are aware that RFC2544 requires testing also with 256 different destination networks. (But currently we do not support it.)
- The connection tracking table of the stateful NATxy is large enough to store all connections defined by the different source port number destination port number combinations.

How to ensure extreme situation 1?

- Use all different source port number destination port number combinations in the preliminary phase *and*
- Set the UDP timeout of the NATxy gateway to a value higher than the length of the preliminary phase.

Note: the maximum connection establishment rate measurement is performed in the preliminary phase.

How to ensure extreme situation 2?

- Use all different source port number destination port number combinations in the preliminary phase *and*
- Enumerate all the possible source port number destination port number combinations in the preliminary phase *and*
- Set the UDP timeout of the NATxy gateway to a value higher than
  - the length of the preliminary phase *plus*
  - the gap between the two phases *plus*
  - the length of the real test phase.

RFC 4814 REQUIRES pseudorandom port numbers

- Our experience with iptables shows that if the connection tracking table is filled using port number enumeration in increasing order, then the maximum connection establishment rate of iptables degrades significantly compared to its performance using pseudorandom port numbers.
- Pseudorandom all different source port number destination port number combinations may be computing efficiently generated by preparing a random permutation of the previously enumerated all possible source port number destination port number combinations using Dustenfeld's random shuffle algorithm.
- The enumeration of the source port number destination port number combinations in increasing or decreasing order (or in any other specific order) MAY be used as an additional measurement.

Request for feedback

- What do you think of the recommended methods?
  - Do they provide meaningful and reasonable results?
  - Will the results satisfactorily characterize the performances of the NATxy gateways?
  - Is there anything missing?
  - Do we need to measure anything about the connection tear down performance?
  - If yes, are *aggregate measurements** usable? (As we cannot do anything else.)

    *Full depletion time of the connection tracking table (using tests with different sizes of tables).