Network Emulator in Linux

Using Linux NetEm to emulate a network



Delay/Jitter



Packet Loss



Corruption



Duplication



Reorder



Bandwidth

- Base for all commands
- tc qdisc add dev <dev> root netem <command>
- tc qdisc replace dev <dev> root netem <command>
- Deleting all NetEm directives
- tc qdisc del dev <dev> root netem
- Showing current queue discipline
- tc qdisc show dev <dev>

Delay/Jitter



Delay/Jitter

- A delay is a latency added to the packet
- A jitter is a variance in the delay over time
- The delay is added by the processing time of routing devices in between
- The jitter is added by different load of the routing devices

Delay/Jitter

- Simulating a delay (in ms) with NetEm
- <base> delay <delay>
- Simulating a jitter (in ms) (together with a delay) with NetEm

<base> delay <delay> <jitter>

• Samples

- ... netem delay 10ms
- ... netem delay 10ms 5ms



- netem supports three loss models
 - Independent loss
 - Markov model
 - Gilbert-Elliott model

- The independent loss
- For every single packet an probability of loss is applied
- Simulating an independent loss with netem

<base> loss random <percentage>

- Samples
- ... netem loss random 2%

- The Markov model and the Gilbert-Elliott model work with states
- The models differentiate between a Gap (or Good) state and a Burst (or Bad) state
- In Burst state all or most packets are lost
- In Gap state all or most packets are transmitted

• The Markov model



- 1 (or G) means Gap or Good (Packet is transmitted)
- 3 (or B) means Burst or Bad (packet is lost)
- 2 means packet is transmitted during burst
- 4 means packet is lost during gap

• The Markov model



- p13 is the probability from G to B
- p31 is the probability from B to G
- etc

- The Gilbert and Gilbert-Elliott model
- G means Gap or Good
- B means Burst or Bad
- p is the probability from G to B
- r is the probability from B to G
- 1-k is the probability of a loss in G
- 1-h is the probability of a transmit in B





• Simulating a loss with a Markov model

<base> loss state <p13> [<p31> [<p32> [<p23> [<p14>]]]

- Samples
- ... netem loss state 1%
- ... netem loss state 1% 5%
- ... netem loss state 1% 5% 10% 50% 2%

• Simulating a loss with Gilbert-Elliott model

<base> loss gemodel [<r> [<1-h> [<1-k>]]]

G 1-k В

1-h

- Samples
- ... netem loss gemodel 1%
- ... netem loss gemodel 1% 5%
- ... netem loss gemodel 1% 5% 10% 10%

Corruption



Corruption

- During processing a packet can get corrupted
- Simulating a certain percentage of corruption

<base> corrupt <percentage>

- Samples
- ... netem corrupt 1%

Duplication



Duplication

- During processing a packet can get duplicated (e.g. resent)
- Simulating a certain percentage of duplication

<base> duplicate <percentage>

- Samples
- ... netem duplicate 1%

Reorder



Reorder

- During processing two packets can be reordered
- Reorder only works if also a delay is specified
- Simulating a certain percentage of reorder

<base> reorder <percentage>

- Samples
- ... netem delay <delay> reorder 25%
- This example sends 25% of the packages directly and 75% are delayed

Bandwidth



Bandwidth

- A (symmetric) bandwidth limitation can also be emulated
- Additionally a repackaging of data can be emulated (e.g. other MTU)
- Simulating a bandwidth limitation

<base> rate <bandwidth> [<overhead> [<cellsize>
[<celloverhead>]]]

- Samples
- ... netem rate 1Mbit
- ... netem rate 1Mbit 5
- ... netem rate 1Mbit 5 128

References

- http://voiptroubleshooter.com/indepth/burstloss.html
- http://voiptroubleshooter.com/indepth/jittersources.html
- https://www.net.t-labs.tu-berlin.de/papers/HH-GEMPLRTSI-08.p df