DNEmu: An Architecture for Distributed Network Emulation

Hajime Tajaki*1, Hitoshi Asaeda*2

*1 National Institute of Information and Communications Technology (NICT), Japan
*2 Keio University

Abstract

In this work, we propose the Distributed Network Emulator (DNEmu) that exploits existing MPI-based network simulation for the requirements of a planetary-scale network experiment. DNEmu addresses the issue of real-time execution with message synchronization through distributed processes, and enables us to evaluate protocols with actual background traffic using a fully controlled distributed environment. Through evaluation with micro-benchmarks, we find that our DNEmu prototype implementation is similar in terms of packet delivery delay and throughput to the existing non-virtualized environment. We also present a use-case of our proposed architecture for a large distributed virtual machine service in a simple control scenario involving actual background traffic on the global Internet. DNEmu will contribute to research in protocol evaluation and operation in a huge network experiment without interfering with the existing infrastructure.

Micro-Benchmarking

- two nodes located at two places, route information is exchanged by zebra/quagga bpg
- Pure linux ping6/perf
- CORE distributed emulator via GRE tunnel, ping6/perf
- DNEmu with ns-3 via GRE tunnel, ping6/perf

Use-Case (on Distributed VMs)

Distributed emulation using a Cloud service (WIDE cloud) considering
- background traffic between distributed sites
- completely reproducible handoff experiment by NEMO over Wi-Fi
- NEMO, BGP and OSPFv3 by Direct Code Execution

Related Work

NEPI: a general framework for network experiments with Python programming interface. Targeting on using multiple testbeds (ns3, Planetlab, Emulab, ORbit, etc) simultaneously.
CORE: a distributed emulator based on light-weight virtualization in kernel space (netns/IMUNES). Thus the performance is good (shown above results). DNEMU focuses on a “single toolset” of the software with a built-in functionality of ns-3. Thus, can use transparently with existing ns-3 simulation models.