Network Simulations with the ns-3 Simulator

Thomas R. Henderson
Department of Electrical Engineering
University of Washington
Seattle, Washington
98195-2500
tomhend@u.washington.edu

Mathieu Lacage
Planete, INRIA
2004 route des lucioles, BP 93
06902 Sophia Antipolis Cedex
FRANCE
mathieu.lacage@inria.fr

George F. Riley
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332-0250
riley@ece.gatech.edu

ABSTRACT

We provide a demonstration of the emerging *ns-3* discrete-event network simulator. Intended to eventually replace the popular *ns-2* simulator, *ns-3* has been under development for over two years, and the initial stable release is scheduled for June 2008. We aim to provide Sigcomm attendees with a sense for what is new in *ns-3* that may help researchers with their future research.

Categories and Subject Descriptors: I.6 [Simulation and Modeling]: Applications

General Terms: Design, Experimentation

1. NS-3 OVERVIEW

The *ns-2* simulator [1] has long been a widely used simulator for research and education on Internet and other network systems. However, work is progressing on a replacement for *ns-2*. Borrowing concepts and implementations from several open source simulators including *ns-2*, yans [2], and GTNetS [3], *ns-3* differs from *ns-2* in several ways, including:

- **new software core:** designed to improve scalability, modularity, coding style, and documentation, the core is written in C++ but with an optional Python scripting interface (instead of OTcl). Several C++ design patterns such as smart pointers, templates, callbacks, and copy-on-write are leveraged. Object aggregation capabilities enable easier model and packet extensions;
- **attention to realism:** the Internet nodes are designed to be a more faithful representation of real computers, including the support for key interfaces such as sockets and network devices, multiple interfaces per nodes, use of IP addresses, and other similarities;
- **software integration:** an architecture to support the incorporation of more open-source networking software such as kernel protocol stacks, routing daemons, and packet trace analyzers, reducing the need to port or rewrite models and tools for simulation;
- **support for virtualization:** lightweight virtual machines running over a (possibly wireless) simulation network are an attractive combination for current research; *ns-3* plans to support a few modes of such operation including a native "process" environment where Posix-compliant applications can be easily ported to run in simulation space with their own private stack, and including support for tying together virtual machines of various types;
- **testbed integration:** *ns-3* will enable the testbed-based researcher to experiment with novel protocol stacks and emit/consume network packets over real device drivers or VLANs. The internal representation of packets is network-byte order to facilitate serialization;
- **attribute system:** researchers require a means to identify and possibly reassign all values used to configure parameters in the simulator. *ns-3* provides an attribute system that integrates the handling and documentation of default and configured values; and
- **tracing architecture:** *ns-3* is building a tracing and statistics gathering framework using a callback-based design that decouples trace sources from trace sinks, enabling customization of the tracing or statistics output without rebuilding the simulation core.

Like *ns-2*, *ns-3* is open-source and licensed under GNU GPLv2, and welcomes developers and contributed code from across academia, industry, and government.

2. ACKNOWLEDGMENTS

The demonstration is based on the collective work of many *ns-3* software developers; space precludes listing them all. This material is based upon work supported by the National Science Foundation under Grant Nos. CNS-0551686, CNS-0551378, and CNS-0551706.

3. ADDITIONAL AUTHORS

Craig Dowell (University of Washington, email: `craigdo@ee.washington.edu`) and Joseph B. Kopena (Drexel University, email: `tjkopena@cs.drexel.edu`)

4. REFERENCES

[1] S. McCanne and S. Floyd. The LBNL network simulator. Software on-line: http://www.isi.edu/nsnam, 1997. Lawrence Berkeley Laboratory.

[2] M. Lacage and T. R. Henderson. Yet another network simulator. In *WNS2 '06: Proceeding from the 2006 workshop on ns-2: the IP network simulator*, page 12, New York, NY, USA, 2006. ACM.

[3] G. F. Riley. Large Scale Network Simulations with GTNETS. In *Proceedings of the 2003 Winter Simulation Conference*, 2003.

SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.
ACM 978-1-60558-175-0/08/08.