Building the Feasible Simulation Platform Supported with Multiple Versions of ns-2

Shih-Tsung Liang

Taipei Municipal University of Education

stliang@ieee.org

Abstract—Network simulator version 2 (ns-2) has being one of the most popular software package for the simulation and performance analysis of network protocols. Due to the open-source policy adopted in ns-2, many researchers have contributed their extensions to ns-2 for public users to patch. Since the patches are based on a diversity versions of ns-2, having multiple versions of ns-2 in hand is highly demanded for ns-2 users. In this paper, we propose to build the feasible simulation platform supported with multiple versions of ns-2 in the Windows XP. In particular, a successful installation of ns-2 with versions 2.26 through 2.34 has been verified by following the proposed procedure. With the platform, plenty of released extensions for different versions of ns-2 can be easily augmented and shared to promote research in areas of communication networks.

Keywords: Network simulator version 2 (ns-2), simulation, performance analysis

I. Introduction

Network simulator version 2 (ns-2) [1] has being one of the most popular software package for the simulation and performance analysis of network protocols. The open-source policy adopted in ns-2 facilitates researchers to contribute their extensions to public users by software patching [1]. With these patches, experiences and results from many researchers can be shared and reused to promote efficient research works in related areas. A common setup of ns-2 is to install ns-allinone package under the cygwin [2] in Windows XP. The installation for the newly released version is quite straightforward. Unfortunately, lots of released patches work only for some specific versions of ns-2. Therefore, having multiple versions of ns-2 co-exist in the system is highly demanded and cumbersome for ns-2 users. The installation overhead comes from the different versions of coordinated software package bundled in the ns-allinone releases as shown in Table 1. In this paper, we come up with the construction of ns-2 based simulation platform supporting multiple versions. In particular, a successful installation of ns-2 with versions 2.26 through 2.34 has been verified by applying the proposed procedure.

Table 1 Summary of software package versions in ns-allinone-2.XX .

NSVER	Ns-2.34	ns-2.33	ns-2.32	ns-2.31	ns-2.30	ns-2.29	ns-2.28	ns-2.27	ns-2.26
TCLVER	8.4.18	8.4.18	8.4.15	8.4.14	8.4.13	8.4.11	8.4.5	8.4.5	8.3.2
TKVER	8.4.18	8.4.18	8.4.15	8.4.14	8.4.13	8.4.11	8.4.5	8.4.5	8.3.2
OTCLVER	1.13	1.13	1.13	1.13	1.12	1.11	1.9	1.8	1.0a8
TCLCLVER	1.19	1.19	1.19	1.19	1.18	1.17	1.16	1.15	1.0b13
NAMVER	1.14	1.13	1.13	1.13	1.12	1.11	1.11	1.10	1.9
XGRAPHVER	12.1	12.1	12.1	12.1	12.1	12.1	12.1	12.1	12.1
ZLIBVER	1.2.3	1.2.3	1.2.3	1.2.3	1.2.3	1.2.3	1.1.4	1.1.4	1.1.4
X11	1.4	1.4	1.3	1.3	1.3	1.3	1.3	1.2	1.2
DEI80211MRVER	1.1.4	1.1.4	N/A	N/A	N/A	N/A	N/A	N/A	N/A

The rest of the paper is organized as follows: Section 2 presents the one-by-one and all-versions-in-one procedures for the bare installation of the simulation platform. Section 3 demonstrates the feasibility to extend the platform from by the released patches dedicated for specific versions of ns-2. Finally, concluding remarks are given in Section 4.

II. The bare platform

The system architecture of the simulation platform supported with multiple versions of ns-2 is depicted in Figure 1. In the following, two scenarios to setup the bare platform are described. In case when the ns-2.34 has been installed, the one-by-one setup can be applied. On the other hand, for a clear setup, the simple all-versions-in-one installation can be adopted.

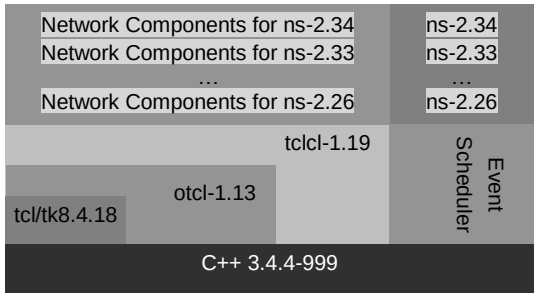

Figure 1 *System architecture of the simulation platform.*

Table 2 *Files to be included in the related configuration file set.*

Configuration file \ ns version	2.33	2.32	2.31	2.30	2.29	2.28	2.27	2.26
ns-2.XX/conf/configure.in.tcl	×	o	o	o	o	o	o	o
ns-2.XX/conf/configure.in.tk	×	o	o	o	o	o	o	o
ns-2.XX/conf/configure.in.otcl	×	×	×	o	o	o	o	o
ns-2.XX/conf/configure.in.TclCL	×	×	×	o	o	o	o	o
ns-2.XX/conf/configure.in.z	×	×	×	×	×	o	o	o
ns-2.XX/conf/configure.in.x11	×	o	o	o	o	o	o	o

A. One-by-one setup

Assume that the ns-2.34 has been installed in the default path. To upgrade the simulator to be supported with multiple versions of ns-2, the following steps are taken:

1 Make a symbolic link to the ns-2.34 execution file ~/ns-allinone-2.34/ns-2.34/ns.exe:

```
$cd ~/ns-allinone-2.34/bin
$ln -s ../ns-2.34/ns.exe -T ns234.exe
```

2 Download ns-2 source files of different versions (i.e., ns-src-2.XX.tar.gz, XX=26~32 and ns-2.33.tar.gz) to directory ~/ns-allinone-2.34/.

3 For each version of ns-2 sources (say version 2.XX, XX=26~33),

① Unzip the source file as follows:

```
$cd ~/ns-allinone-2.34
$tar xvfz ns-src-2.XX.tar.gz
```

② For pre-2.28 versions of ns-2 (i.e., XX=26 or XX=27), some files in directory ns-2.XX may need to be modified to conform to new version of gcc bundled in the newly released cygwin. Check your gcc version as follows:

```
$ cygcheck –c gcc
Cygwin Package Information
Package   Version     Status
gcc       3.4.4-999   OK
```

If your gcc version is greater than gcc-3.2, some source codes should be modified to fix the compilation problems. To share our experience, patch files for ns-2.26 and ns-2.27 conforming to gcc-3.4.4 is given in files ns226-gcc344.patch and ns227-gcc344.patch [3], respectively. Download ns2XX-gcc344.patch to directory ~/ns-allinone-2.34 and patch the file as follows:

```
$cd ~/ns-allinone-2.34/
$patch -p0 < ns2XX-gcc344.patch
```

③ Update the related configuration files:

```
$cd ~/ns-allinone-2.34/ns-2.XX/conf
```

For each file (say file F) in the related configuration file set (refer to Table 2), run the shell command:

```
$cp ../../ns-2.34/conf/F .
```

When all the files are updated, re-configuration should be done as follows:

```
$cd ..
$autoconf
```

④ Compile to build the execution file

```
./configure
make clean
make
```

⑤ Make a symbolic link to ns.exe

```
cd ~/ns-allinone-2.34/bin
ln -s ../ns-2.XX/ns.exe -T ns2XX.exe
```

When all have been done, you should enter ns2XX to run ns in version 2.XX. The installed file structure is shown in Figure 2.

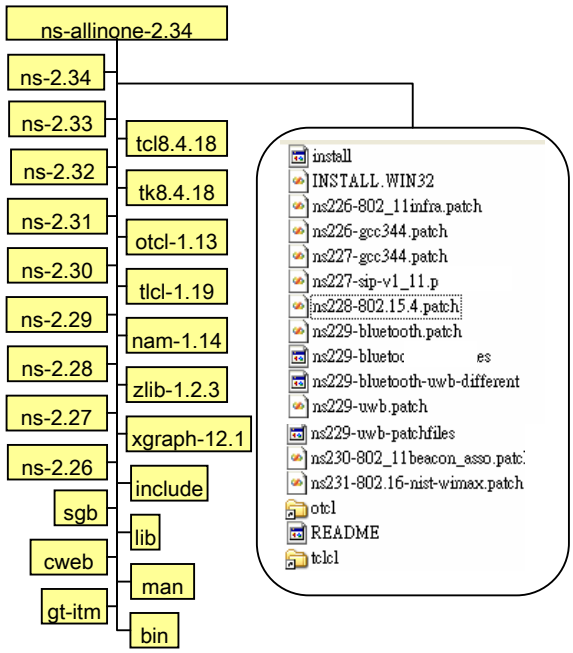

Figure 2 *File structure of the simulation platform.*

B. All-versions-in-one installation

To facilitate the installation of the simulation platform, we come up with the all-versions-install.patch for ns-2 users. With the patch, only four steps to complete the installation of the simulation platform under the cygwin environment:

1 Download ns-allinone-2.34.tar.gz and unzip the file as follows:

```
cd ~
tar xvfz ns-allinone-2.34.tar.gz
```

2 Download ns-2 source files of different versions (i.e., ns-src-2.XX.tar.gz, XX=26~32 and ns-2.33.tar.gz) to directory ~/ns-allinone-2.34/ and for each version of ns-2 unzip the source file as follows:

```
cd ~/ns-allinone-2.34
tar xvfz ns-src-2.XX.tar.gz
tar xvfz ns-2.33.tar.gz
```

3 Download and applying all-versions-install.patch as follows:

```
cd ~
patch –p0 <all-versions-install.patch
```

4 Install the bare platform supported with multiple versions of ns-2:

```
cd ~/ns-allinone-2.34
./install
```

III Extension from released patches

The ns-2 simulator has become one of the de facto software packages for the simulation and performance evaluation of network protocols. Many research works in areas of communication networks have been based on it. Due to the open-source model, easy extension of ns-2 to support new protocols can be achieved. In recent years, a number of patches from researchers with a variety of expertise have been released. By sharing and re-using these extending patches, the research and development capacity for designing network protocols can be improved. Refer to the ns-2 official website for seeking the contributed modules. A number of released patches for various versions of ns-2 in the mobile and wireless network area are summarized in Table 3.

To simulate with any desired extension, users need to follow the setup procedure attaching to the released package. However, a variety of procedures have been adopted for existing releases. To unify and simplify the setup procedure, we also re-produce patch files [3] for these extensions by using the utility program *diff* with options *–uNr*. For example, based on the original IEEE 802.15.4 MAC extension for ns-2.28 through updating files, we have came up with the patch file — ns228-zigbee.patch as follows:

1 Set up the working directories

```
$cd ~
$tar vxzf ns-src-2.28.tar.gz
$mv ns-2.28 ns-2.28-orig
$tar vxzf ns-src-2.28.tar.gz
$mv ns-2.28 ns-2.28-zigbee
```

2 Update files in directory ns-2.28-zigbee based on the original release[10].

3 Generate the patch file:

```
$diff –uNr ns-2.28-orig ns-2.28-zigbee > ns228-zigbee.patch
```

In the same manner, existing releases can be represented as their corresponding patch files. With such patch files, the simple and normalized procedure can be adopted for the extension of the bare platform. As an example, we can extend the platform to support simulation of Bluetooth networks as follows:

1 Copy the patch file (say ns229-bluetooth.patch) to be under directory ns-allinone-2.34.

2 Apply the patch:

```
cd ~/ns-allinone-2.34
patch –p0 <ns229-bluetooth.patch
```

3 Re-compile:

```
cd ns-2.29
./configure
make clean
make
```

When another extension is needed hereafter, un-patching the previous extending module and patching the new one is a sound solution regardless

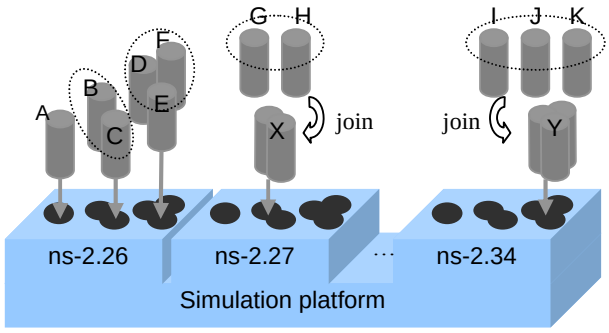

Figure 3 *Extending the bare platform by patching.*

Table 3 Some released ns-2 extensions in the area of wireless networking.

Extending module	ns-2 version	Description	Re-produced unified patch [3]
IEEE 802.11s wireless mesh networks [4]	Ns-2.33	Extend ns-2 to partially support IEEE 802.11s D/3.0. Backward compatible to the original 802.11 module with additional support of wireless distribution system.	http://tmue.edu.tw/~stliang/release/ns2patches/ns233-802_11s.patch
IEEE 802.16 WiMax (MAC+PHY) [5]	ns-2.31	Extend ns-2 to partially support IEEE 802.16 MAC and Phy with features including 1)Wireless MAN-OFDM physical layer with configurable modulation; 2)Time Division duplexing (TDD); 3)Management messages to execute network entry (without authentication); 4)Default scheduler providing round robin uplink allocation to registered Mobile Stations (MSs) according to bandwidth requested; 5) IEEE 802.16e extensions to support scanning and handovers; and 6) Fragmentation and reassembly of frames. The source bundle with ns-allinone-2.31 is available on request.	http://tmue.edu.tw/~stliang/release/ns2patches/ns231-802.16-nist-wimax.patch
IEEE 802.11 infrastructure mode [6]	ns-2.30	Extend ns-2 to partially support IEEE 802.11 infrastructure mode. Beacon, Passive Scanning, and Association functions are implemented. The patch for ns-2.30 is available.	http://tmue.edu.tw/~stliang/release/ns2patches/ns230-802_11beacon_asso.patch
UWB wireless networking [7,8]	ns-2.29	Extend ns-2 to partially support the Ultra Wide-Band wireless networking. In particular, the DCC-MAC layer and the impulse radio UWB physical layer are implemented. The source bundle with ns-allinone-2.29 is available.	http://tmue.edu.tw/~stliang/release/ns2patches/ns229-uwb.patch
BlueTooth [9]	ns-2.29	Bluetooth extension for NS2 at the University of Cincinnati implements a full bluetooth stack, including Baseband, LMP, L2CAP, BNEP layers. The extension follows spec 1.1 and is partially updated to spec 2.0. The source bundle with ns-allinone-2.29 is available.	http://tmue.edu.tw/~stliang/release/ns2patches/ns229-bluetooth.patch
IEEE 802.15.4 MAC (ZigBee) [10]	ns-2.28	Extend ns-2 to partially support the contention access part (CAP) of IEEE 802.15.4 MAC. The files changed for ns-2.28 are available.	http://tmue.edu.tw/~stliang/release/ns2patches/ns228-802.15.4.patch
SIP module [11]	ns-2.27	Extend ns-2 to simulate SIP signaling. Key features included are 1)Layered structure; 2)Stateful entities; 3)100rel SIP extension; 4)RecordRoute/Route; and 5)considering processing delays. The patch for ns-2.27 is available.	http://tmue.edu.tw/~stliang/release/ns2patches/ns227-sip-v1_11.patch
IEEE 802.11 infrastructure mode/ with handoff [12]	ns-2.26	Extend ns-2 to partially support IEEE 802.11 infrastructure mode. Beacon, Passive Scanning, Association, Re-Association functions and the wired distribution system through 802.1D bridging (not the IEEE 802.11f) are implemented. The patch for ns-2.26 is available.	http://tmue.edu.tw/~stliang/release/ns2patches/ns226-802_11infra.patch

the dependency between these two patches. In occasions, however, ns-2 users need to simulate by combing multiple released extensions for the same version of ns-2. Figure 3 presents the framework to apply multiple patches to the simulation platform. As shown in the figure, when all the required patches are independent, the target simulation platform can be obtained by simply applying patching one by one to the bare platform. However, if two patches depend to each other, manually joining of the two patches should be performed first. For instance, to extend ns-2.29 with both the Bluetooth and Ultra Wide-Band modules, applying the released patches—ns229-bluetooth.patch and ns229-uwb.patch one after another would incur problems. Through running simple shell commands, as shown below, the dependency between the two patches reveals five files including Makefile.in, packet.h, wireless-phy.h, ns-lib.tcl, and ns-packet.tcl should be joined prior to applying patching.

```
$grep “diff –uNr” ns229-bluetooth.patch > ns229-bluetooth
$grep “diff –uNr”  ns229-uwb.patch >ns229-uwb
$diff –u ns229-bluetooth ns229-uwb | grep –h “ diff “
diff -uNr ns-2.29/Makefile.in ns-2.29/Makefile.in
diff -uNr ns-2.29/common/packet.h ns-2.29/common/packet.h
diff -uNr ns-2.29/mac/wireless-phy.h ns-2.29/mac/wireless-phy.h
diff -uNr ns-2.29/tcl/lib/ns-lib.tcl ns-2.29/tcl/lib/ns-lib.tcl
diff -uNr ns-2.29/tcl/lib/ns-packet.tcl ns-2.29/tcl/lib/ns-packet.tcl
```

IV Conclusions

In this paper, we have come up with the building and extending of the simulation platform supported with multiple versions of ns-2. The contribution of this paper is two-fold. First of all, we have provided both the one-by-one and the all-versions-in-one procedures for the installation of the bare simulation platform. In particular, the bare platform supported with ns-2.26 through ns-2.34 has been built to show the feasibility of the proposed procedure. With the platform, plenty of released extensions for different versions of ns-2 can be easily shared between researchers in areas of communication networks. Second, the normalized patching to the bare platform has been achieved through the unified process of patch generation. Based on the process, we have provided the conformable patches for some existing extensions in the area of wireless networking.

References

[1] The VINT Project, The UCB/LBNL/VINT Network Simulator—ns (version 2), Web site: http://nsnam.isi.edu/nsnam/.

[2] Rethat, NU+Cygnus+Windows=Cygwin, Web site: http://cygwin.com/.

[3] Shih-Tsung Liang, Release Notes and Patch Files for Building and Extending the Feasible Simulation Platform supported with Multiple Versions of ns-2, Web site: http://tmue.edu.tw/~stliang/release/multivers4ns2.htm.

[4] Shih-Tsung Liang and Sheng-Lung Lee, "The NS-2 Module for IEEE 802.11s Wireless Mesh Networks," Workshop on Computer Networks and Web Service/Technologies, NCS 2009, November 27-28, 2009.

[5] Richard Rouil, "The Network Simulator NS-2 NIST add-on IEEE 802.16 WiMax (MAC+PHY)." Web site: http://www.antd.nist.gov/seamlessandsecure/doc.html

[6] Ilango Purushothaman, Sumit Roy**,** **"**Infrastructure mode support for IEEE 802.11 implementation in NS-2," Technical Report, FUNLAB, University Washington. Web site: http://ee.washington.edu/research/funlab/projects.htm.

[7] Ruben Merz, Jean-Yves Le Boudec, and Jörg Widmer, "An Architecture for Wireless Simulation in NS-2 Applied to Impulse-Radio Ultra-Wide Band Networks," 10th Communications and Networking Simulation Symposium, Norfolk, VA, March 25-29, 2007. Web site: http://icawww1.epfl.ch/uwb/ns-2/.

[8] Ruben Merz, Jörg Widmer, Jean-Yves Le Boudec, and Bozidar Radunovic, "A Joint PHY/MAC Architecture for Low-Radiated Power TH-UWB Wireless Ad-Hoc Networks," Wireless Communications and Mobile Computing Journal, Special Issue on Ultrawideband (UWB) Communications, vol. 5, num. 5 (2005), p. 567-580, 2005. Web site: http://icawww1.epfl.ch/uwb/ns-2/.

[9] Dharma Agrawal and Qihe Wang, UCBT - Bluetooth extension for NS2 at the University of Cincinnati, Web site: http://www.ececs.uc.edu/~cdmc/ucbt/ucbt.html.

[10] Iyappan Ramachandran, IEEE 802.15.4 MAC implementation in NS-2. Web site: http://www.ee.washington.edu/research/funlab/802_15_4/.

[11] Rui Prior, A SIP module for ns-2.27. Web site: http://www.ncc.up.pt/~rprior/ns/.

[12] Shih-Tsung Liang, Chung-Yi Wu, and Ming-Yi Weng, "Design and Implementation of Network Simulator for IEEE 802.11 Infrastructure Wireless LANs," in Proc. 10th Mobile Computing Workshop, March 2005. Web site: http://tmue.edu.tw/~stliang/.