Network Modeling and Emulation: A New Architecture of Next Generation WANem

Swaswati Goswami[1], Hemanta Kumar Kalita, Munmi Upadhyay, Bijoy Sankar Rajkumar and Kanhaiya Thakur

Dept of Information Technology, School of Technology

North Eastern Hill University, Shillong

India–793022

[1]Email: swaswati6@gmail.com

Abstract: *Typically application developers develop and testing team tests an web based application in a local area network (LAN). And, once everything is completed deployment of the application is done in a wide area network (WAN). This process carries lots of WAN related bugs to the application. This is due to LAN having higher bandwidth (for example, 100 Mbps), lesser delay (~1 ms), almost zero loss, corruption and reordering of packets as compared to WAN. WANem[8] is used to provide real experience of a WAN during application development or testing over LAN environment. Thus, WANem helps both application developers and testers by emulating WAN in a LAN. Next Generation WANem (NxGen WANem) is developed based on existing WANem with enhanced features. With more interactive GUI and modifications on some preexisting commands, the NxGen WANem provides user with facility to model and design a network by dragging the network components available in the interface designed and thereafter emulate the network characteristics. In this paper, we propose an architecture of NxGen WANem.*

Keywords: Emulation, LAN, WAN, WANem, SWT, GWT, JSP, AJAX

1. Introduction

In Figure 1 we give architecture of WANem[9][11][10][8], a Wide Area Network emulator tool.
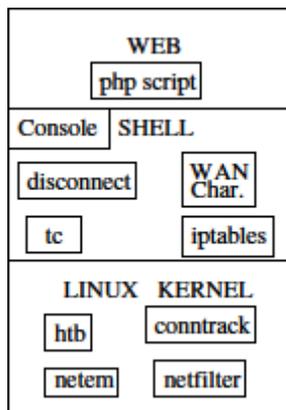
Figure 1. Architecture of WANem

WANem has three layers: web, shell and linux kernel. In the bottom layer, WANem works on linux kernel and uses modules like htb, conntrack, netem and netfilter. In the middle layer, using shell script WANem console is developed. Some of the commands which are available in WANem console are–reset, nat, shutdown, restart, status, wanemreset, assign and exit2shell. WANem also implements features like disconnection, WAN characterization and uses existing features like tc and iptables in the middle layer. Top layer of WANem is a web based GUI designed using PHP script.

WANem GUI basically has three features. They are– WANalyzer, basic mode, and advanced mode. WANalyzer is a WAN characterization tool. In basic mode WANem can emulate delay and bandwidth. The advance mode has the capacity to emulate bandwidth, delay, packet loss, duplication, packet reordering, corruption, disconnection etc. In Figure 2 WANem GUI in advanced mode is shown.

Figure 2. WANem GUI

It may be noted that WANem is released to open source and one can download it from sourceforege.net/wanem. Use and setting of WANem in LAN is well described in its official website[12]. Tutorial on WANem is also available in the

Internet. For example, openmaniak.com. Current version of WANem allows user to enter and execute WANem commands using web based GUI sitting from anywhere in a LAN. This is due to PHP script running on Apache web server in the WANem box. However, in current version of WANem a user can't model and design a WAN in the GUI itself. A user first of all model and design a WAN using pen and paper. Then he enters the relevant information such as WAN parameters, IP addresses, and port etc in the WANem GUI. This is painful as a user does not see overall picture of the network he is emulating. It would be nicer if we could model and design the WAN using drag-and-drop of network elements such as WAN cloud, client, server and link in the GUI itself. In NxGen WANem we add this new feature. Here, a user shall be able to model and design a WAN using a web based interactive GUI and there on apply WANem commands just with the click of a button.

Remainder of the paper is divided into four sections. In Section 2 we give literature survey. In Section 3 NxGen WANem architecture, design and algorithm are proposed. Section 4 discusses implementation and results. Finally, in Section 5 we conclude the paper.

2. Literature survey

Some of the network emulators have been used in the past for different purposes and using different platform. Among them Emusocket[1] is a java based network emulator. NISTNet[13] and netem[6] are open source network emulator tools developed in Linux. NISTNet and netem are widely used tools which solve our purpose. But they are not easy to use and deploy. WANem works on top of netem and hence is dependent on netem. As stated above, WANem's user interface has limitations. The network characteristics are given as input through text boxes which as a result does not provide a flexible and sophisticated GUI. To overcome this problem NxGEN WANem seems to be a unique solution with a sophisticated and user friendly GUI which can extend the capability of a user to model and design his own wide area network.

Authors originally thought of using SWT (Standard Widget Toolkit)[14] for the purpose of designing the NxGen WANem GUI. SWT is an open source widget toolkit for Java, designed to provide efficient, portable access to the user-interface facilities of the operating systems on which it is implemented, to display GUI elements. Though SWT is an useful API for desktop applications, yet it is found that SWT is not easy to implement with web based applications.

A parallel study on GWT (Google Web Toolkit)[3] along with SWT was also carried out. GWT is based on Java and is not trivial to setup. Also, there is not much good documentation and support community available for GWT. While studying various drag and drop features provided by GWT, authors came across jQuery-ui API[7]. jQuery-ui API can be used to

incorporate the drag and drop facilities to the network elements such as PC, link, WAN cloud etc. The API has the following interactions: draggable, droppable, resizable, selectable and sortable. jQuery is very simple to setup. All we need to do is download the jquery.js file from Internet and add the packages jquery-1.9.1.js and jqueryui.js.

3. Proposed NxGen WANem

The proposed framework for NxGEN WANem is a modification of the previous WANem in both architectural level as well as the user interface level. Although, the major changes occur in the web based layer, Figure 3 shows the modifications of the other two layers as well.

3.1 Architecture

The architecture of the NxGEN WANem is as shown in Figure 3. As can be seen from the figure, the bottom and middle layer of NxGen WANem is almost similar to WANem as only the scripting language in the shell is changed to awk and the netem module in the linux kernel is renamed as wenem. Top layer, which is web based layer is what is different.
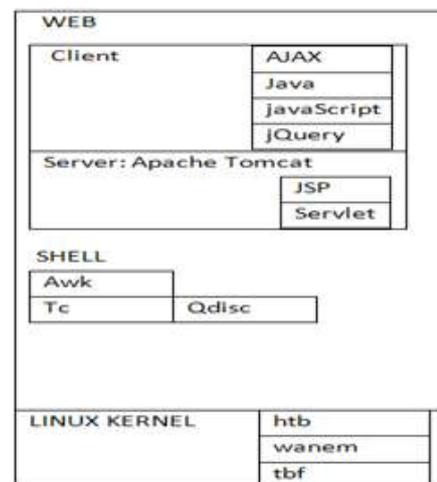
Figure 3. Architecture of NxGEN WANem

A typical user generally interacts with NxGen WANem using web based layer. The changes in this layer are vital. The server side scripting language is changed to JSP and servlet from php and the Apache web server is replaced with Apache Tomcat in NxGen WANem. To save the network characteristics and interact with the client, server side technologies such as JSP and servlets are used. In the client side, different technologies such as AJAX[5], [15], Java, JavaScript[4] and JQuery[2] are used to design a GUI with ActiveX Components. These ActiveX components can be dragged and dropped to design a network. The previous architecture of WANem is lacking of such a user friendly GUI.

3.2   Design

NxGen WANem GUI design depicts the modifications in the web based layer. GUI provides a means to the user to design a network with lesser complexity. Here, we discuss the implementation of the GUI in detail. The design consists of the detailed use of the properties of AJAX, jquery and javascript. The GUI as shown in Figure 4 consists of the following parts:

- Menu Bar. A menu bar, with menu items such as About, WANalyzer, Apply Settings, Save/Restore, Remote Terminal etc.
- Left Panel. Left panel contains network elements (ActiveX components) such as PC, WAN Cloud and Link with drag-and-drop and resizable feature.
- Design Panel. A design panel or tray is in the middle of the GUI for designing the network.
- Property Window. Right panel, called the property window consists of text fields and buttons which enables a user to enter various properties of a network element and also to apply the settings. Note that Apply Setting button is used to run WANem command in the background.

Figure 4. NxGen WANem GUI

The properties of the network elements (ActiveX components) designed using client side scripting language are:

- PC: drag, drop, clone
- WAN Cloud: drag, drop, clone
- Link: drag, drop, resizable

While designing the left panel with the help of AJAX, JavaScript and JQuery, the Steps for adding various components in the NxGen WANem GUI and how they interact are described below:

1) For PC, WAN Cloud and Link icon, helper is set as clone so that the icon can be cloned and containment is set as tray which means the icon can be dropped in the <iframe> named tray. Note that helper and containment are the options of the event ui.draggable of a function in jquery-ui.js package.
2) For Link icon event ui.resizable which is an widget of the function in jquery-ui.js is used to make the link

resizable. Again handles is set as e, w so that the Link icon can be extended in both east and west (right and left) directions.
3) Again, a dragstop event is bound to each of the draggable components (PC, WANCloud, Link) so that the components are dropped while the mouse button is released.
4) When the PC icon or WANCloud icon is dragged from the left panel and added to the tray, properties window prompts a user to enter the properties corresponding to the PC or WANCloud (for example, IP address, Name, Loss, Duplicate, Corruption, Reorder etc).
5) Once the user enters the corresponding properties, a save button appears. The save button for PC, WANCloud, Link are identified by the identifiers pcButton, wanButton, and linkButton respectively. Initially these buttons are kept hidden using the function hide().
6) When an event change such as mouse up in the textfields occur, validation check for empty fields are done and once the validation check returns true, save button appears.

3.3   Proposed Algorithm

The implementation of NxGEN WANem is distributed in various parts i.e. modification of the kernel module, designing the GUI and re-mastering of knoppix. In this paper, however we concentrate on NxGen WANem GUI. In Algorithm–1,2 and 3 we propose pseudo codes for NxGen WANem GUI.

4.   Implementation and results

In Figure 5 we show the implemented NxGen WANem GUI. The network elements (computer, link, WANcloud) can be dragged from the left panel and dropped in the middle frame. While any of these components are dropped in the frame window, the corresponding properties of that component appear in the right panel. The various properties of different components are:

- Computer: Name, IP
- WANcloud: Name, IP, Delay, Loss, Duplicate, Reorder, Corruption
-

A user needs to enter the properties. For example, when a user enters the Name and IP address of the computer he/she has dropped then after validating all the fields a save button appears. A sample designed network is shown in Figure 6. After applying Delay the result can be viewed as shown in Figure 7.

```
1 begin
       // Import necessary header files
       // HTML page, head and javascript
2      Function() begin
           // Make the PC draggable
3          Set helper:clone
4          Set Containment:tray
5          Set Label PC name→visible
6          Set Label IP address→visible
7          Set Text field PC name→visible
8          Set Text field IP address→visible
           // Make the Link draggable
9          Set helper:clone
10         Set Containment:tray
           // Make the Link resizable
11         Set handles:e, w
12         Set Label direction→visible
13         Set Text field Link direction→visible
           // Make the WANCloud draggable
14         Set helper:clone
15         Set Containment:tray
16         Set Label WANem PC name→visible
17         Set Label WANem PC IP address→visible
18         Set Label delay→visible
19         Set Label loss→visible
20         Set Text field WANem PC name→visible
21         Set Text field WANem PC IP
           address→visible
22         Set Text field delay→visible
23         Set Text field loss→visible
24     end
25 end
```

Algorithm 1: NxGen WANem GUI: Pseuodocode Part–1.

```
1 begin
       // Create an XMLHttpRequest
2      xmlhttp→new XMLHttpRequest()
3      Function sendLinkMessageToServer() begin
4          xmlhttp.open("POST","ServletName?arg1",
           true)
           // arg1 parameter(Link name)
              to be passed to the Servlet
5          xmlhttp.send()
6      end
7      Function sendPcMessageToServer() begin
8          xmlhttp.open("POST","ServletName?arg1 &
           arg2", true)
           // arg1 and arg2 are the
              parameters(PC name and IP
              address) to be passed to
              the Servlet
9          xmlhttp.send()
10     end
11     Function sendWANMessageToServer() begin
12         xmlhttp.open("POST","ServletName?arg1 &
           arg2 & arg3 & arg4" ,true);
           // arg1, arg2,arg3,arg4 are
              the parameters(PC name,IP
              address, delay,loss)to be
              passed to the Servlet
13         xmlhttp.send();
14     end
15     Initially set
16       Save button for saving PC properties→hidden
17       Save button for saving WANem
         properties→hidden
18       Save button for saving LINK
         properties→hidden
19     If the textfields PCName and IPaddress are
       valid
20       Set Save button for PC→visible
21     If the textfields WANem PCName, IP address,
       delay and loss are valid
22       Set Save button for WANem→visible
23     If the textfields Linkdirection is valid
24       Set Save button for Link→visible
       // End of script, HTML head
25 end
```

Algorithm 2: NxGen WANem GUI: Pseuodocode Part–2

```
1  begin
      // HTML body
2    Set menu bar with menu items About,
     WANalyzer, Apply Settings, Save/Restore,
     Remote terminal
3    Set PC, Link, WANCloud
4    Set <iframe> with id tray which is the
     network design panel
5    Set the property window with text fields and
     Labels
6    On clicking the Save button for PC, call
     sendPcMessageToServer()
7    On clicking the Save button for WANCloud,
     call sendWANMessageToServer()
8    On clicking the Save button for the Link, call
     sendLinkMessageToServer()
9    Click Apply Settings button to apply the settings
     to the WANem PC
      // End of HTML body and Page
10 end
```

Algorithm 3: NxGen WANem GUI: Pseuodocode Part−3

Figure 5. NxGEN WANem GUI

While pinging the applied properties are shown as in Figure 8. Table I shows the applied delay and the delay measured in terms of round trip time(RTT). The table further

Figure 6. NxGEN WANem GUI: Designed network with applied properties
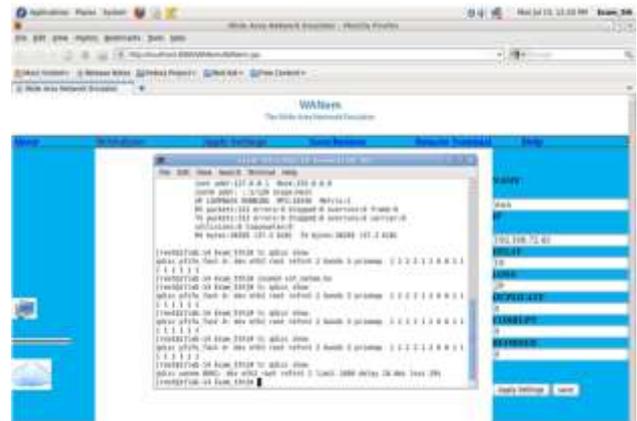
Figure 7. NxGEN WANem GUI: Applied Delay
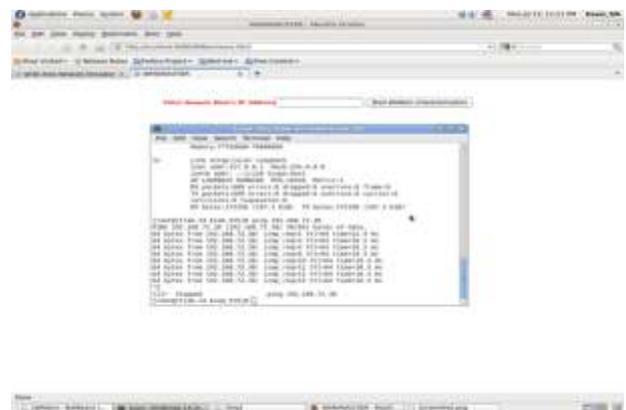
Figure 8. NxGEN WANem: Showing variation in RTT after applying delay shows that the delay measured is almost twice as that of the applied delay as it is measured in terms of RTT.

TABLE 1: TABLE DISPLAYING THE MEASURED DELAY IN A LAN ENVIRONMENT AFTER APPLYNING SOME DELAY THROUGH THE NXGEN GUI

Delay(ms)	
Applied Delay	*Measured Delay*
10	20.4
20	40.2
15	30.2
13	26.4
5	10.3

5. Conclusion

In this paper we have proposed a new architecture of NxGen WANem and discussed in detail about the top layer of the architecture. So far we are able to design a network in

graphical form where we can drag and drop different network elements like client PC, WAN cloud and link. The previous version of WANem was however limited only to specify inputs through text boxes. The NxGen WANem GUI designed works well but the network components are not available as transparent as it should be. We have so far implemented only the limited modes of operation available in WANem like Basic Mode. There are however other areas of works left like implementing Remote Terminal, Advanced Mode and implementing WANAlyzer etc. This is left as future work.

References

[1] A. Marco Avvenuti and Alessio Vecchio. Application-level network emulation: The emusocket toolkit. Network and Computer Applications, 29(4):343–360, 2006.

[2] Bear Bibeault and Yehuda Kats. jQuery in Action. DreamtechPress, 2008.

[3] Ed Burnette. Google web toolkit, 2006.

[4] H David Flanagan. JavaScript: the definitive guide. " O'Reilly Media, Inc.", 2006.

[5] Jesse James Garrett et al. Ajax: A new approach to web applications. 2005.

[6] Stephen Hemminger. Network Emulation with NetEm. Open Source Development Lab, shemminger@osdl.org, 2005.

[7] jQuery Foundation. jquery user interface, 2013.

[8] Hemanta Kumar Kalita and Manoj K. Nambiar. Designing wanem: A wide area network emulator tool. In COMSNETS. IEEE 978-1-4244-8953-4/11, 2011.

[9] Hemanta Kumar Kalita, Manoj K. Nambiar, and Debadatta Mishra. A new algorithm for measuring available bandwidth in a wide area network. In ADCOM'07. IEEE Explore, 2007.

[10] M. Nambiar and H. K. Kalita. Design of a new algorithm for wan disconnection emulation and implementing it in wanem. In Proceedings of the International Conference and Workshop on Emerging Trends in Technology, ICWET '11, pages 376–381, New York, NY, USA, 2011. ACM.

[11] M. K. Nambiar and Hemanta Kumar Kalita. A simple, efficient icmp based method of network characterization. In Int. CMG Conference, pages 645–654. Computer Measurement Group, 2008.

[12] Manoj K Nambiar, Hemanta Kumar Kalita, Debadatta Mishra, and Shirish Rane. Wanem:Wide area network emulator, 2007.

[13] NISTNET. Nist net, 2006.

[14] Steve Northover and Mike Wilson. Swt: the standard widget toolkit, volume 1. Addison-Wesley Professional, 2004.

[15] Linda Dailey Paulson. Building rich web applications with ajax. Computer, 38(10):14–17, 2005.

Author Profile

Swaswati Goswami, M.Tech. She received her bachelor degree in Information technology from North Eastern Hill University (India) in 2013 and master degree in Information technology from Tezpur University in 2015.

Hemanta Kumar Kalita is working as an Associate Professor in the dept of Information Technology, North Eastern Hill University, Shillong.

Munmi Upadhyay, B.Tech. She received her bachelor degree in Information Technology from North Eastern Hill University (India) in 2013. She is working as Senior Operation Professional in IBM.

Bijoy Sankar Rajkumar, B.Tech. He received his bachelor degree in Information Technology from North Eastern Hill University (India) in 2013. He is working as the Associate System Engineer in IBM.

Kanhaiya Thakur, B.Tech. He received his bachelor degree in Information Technology from North Eastern Hill University (India) in 2013. He is currently working as Operation Executive & city head at TCS in ION Assessment Project, Shillong, Meghalaya.