Problems faced in Communicate set up of Coordinator with GUI and Dispatcher in NCTUns network simulator

D.Satish Kumar
Department of Computer Science and Engineering
Anna university of Technology Coimbatore
Tel: +91-9843723396 E-mail: satishcoimbatore@yahcoo.co.in

Dr.N.Nagarajan
Department of Computer Science and Engineering
Coimbatore Institute of engineering and Technology
E-mail: swekalnag@gmail.com

Abstract

Distributed Emulation can be carried out between real world applications with a simulator. NCTUns-6.0 is a simulator based on kernel re-entering and distributed Emulation simulation methodology, with this methodology a more realistic network model can be simulated for real time applications. Coordinator is a Program that sets up communication between the GUI and Dispatcher in NCTUns-6.0 I, a network simulator running on Fedora-13. We have given in this paper how problems faced in communication set up of Coordinator with GUI and Dispatcher are overcomed . This paper focuses on how to overcome the above problems and execute simulations in real time. This paper also deals with KDE, SELINUX, iptables, environment variables and X-server for NCTUns-6.0 simulation engine

Keywords: Coordinator, Dispatcher, nctunsclient, NCTUns-6.0, Fedora-12, SELinux, IPtables

1. Introduction

Network Emulation can be conducted by connection real world devices to a single machine. Distributed emulation is a novel methodology that allows multiple machines to cooperatively conduct an emulation case. By using this methodology, the System resource required to run an emulation case can come from multiple machines (e.g., multiple servers).

NCTUns is a kernel based network simulator that integrates user-level processes, operating system kernel, and the user-level simulation engine into a cooperative network simulation system, To meet the paper limitation limit NCTUns is briefly explained in

The purpose of our research is to find effective and efficient means of communication setup between the the Coordinator and Dispatcher and GUI program.. In this paper, we use some simple basic solutions to solve the problems. We further claim that it will improve the quality and performance of Network Emulations. This paper gives solution for all users of NCTUns, but still there are problems which we are focusing in future.

The paper is organized as follows. In section 2 we describe the brief concept of Distribution Emulation Architecture. In section 3 we portray the Coordinator and Dispatcher and GUI concepts. Section 4 gives the details about the Problems faced and solutions, conclusion and future works are discussed in the last section

2. DistributionEmulation Architecture

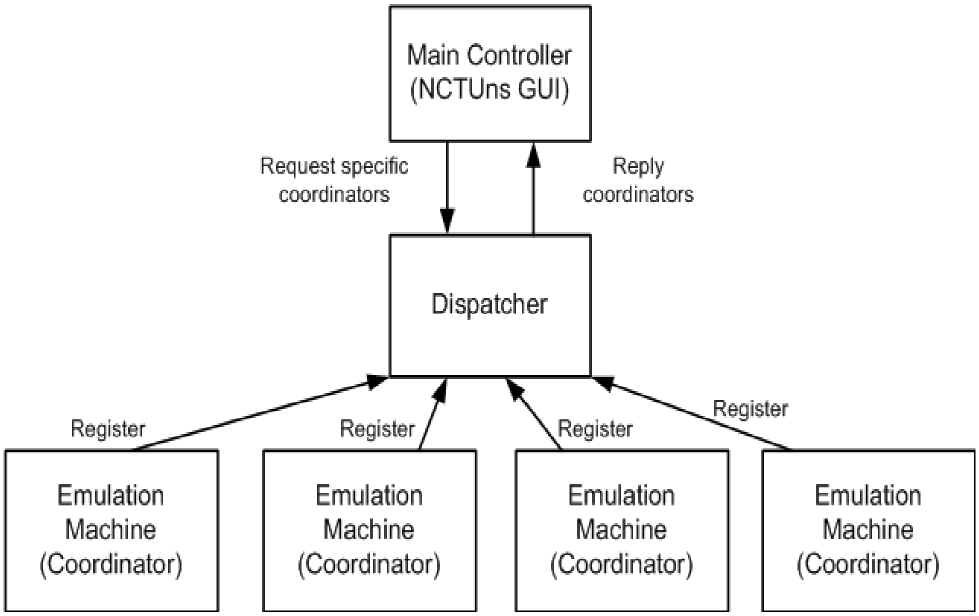

Main Controller (NCTUns GUI)
Request specific coordinators
Reply coordinators
Dispatcher
Register
Register
Register
Register
Emulation Machine (Coordinator)
Emulation Machine (Coordinator)
Emulation Machine (Coordinator)
Emulation Machine (Coordinator)

Fig 1: Distributed Emulated architecture

The Distributed Emulated architecture for NCTUns is shown above , it has a main controller called NCTUns GUI connected to dispatcher , A Dispatcher is a program which connects several emulation machines (coordinator) The main controller manages all the emulated machines .

3 Related Work.

We have refered several papers and web sites to solve all problems occurred between. To meet the paper length limitation, we listed below.

3.1 Definition and Examples

Definition

We first give simple definitions for key terms as they will be used in the rest of this paper:

- *Coordinator***:** Coordinator is a program to communicate with GUI and dispatcher, the coordinator is responsible for reporting the status of simulation study.
- *Dispatcher***:** Dispatcher program can simultaneously manage and use multiple simulation servers[ie, emulation machines] to increase the aggregate simulation throughput. It can be run on a separate machine or on a simulation server.
- *GUI* **:** GUI is graphics user interface , helps the user to edit a network topology and execute it in easy way. It uses TCP/IP sockets to communicate with other components.
- *NCTUnsclient:* It is a program in NCTUns that performs the following six main functions: *Topology Editor*: An environment to specify a network topology Edit Property. *Node Editor*: An environment to configure which modules to be used inside a node, *Simulation Operation Front-End Interface* An interface to control the execution of simulations Packet *Animation Player*::A player to play back logged packet transfer trace *Performance Monitor:* It is for monitoring and used to plot and display logged performance metrics over time. *Retrieve Results*: An interface to view and control submitted background jobs

- *SELinux:* SELinux (*Security-Enhanced Linux*) in Fedora is an implementation of *mandatory access control* in the Linux kernel using the *Linux Security Modules* (LSM) framework. It is set to permissive or enforcing mode.

- *Firewall or IPtables:* IPtables is administration tool / command for IPv4 packet filtering and NAT. You need to use the following tools:

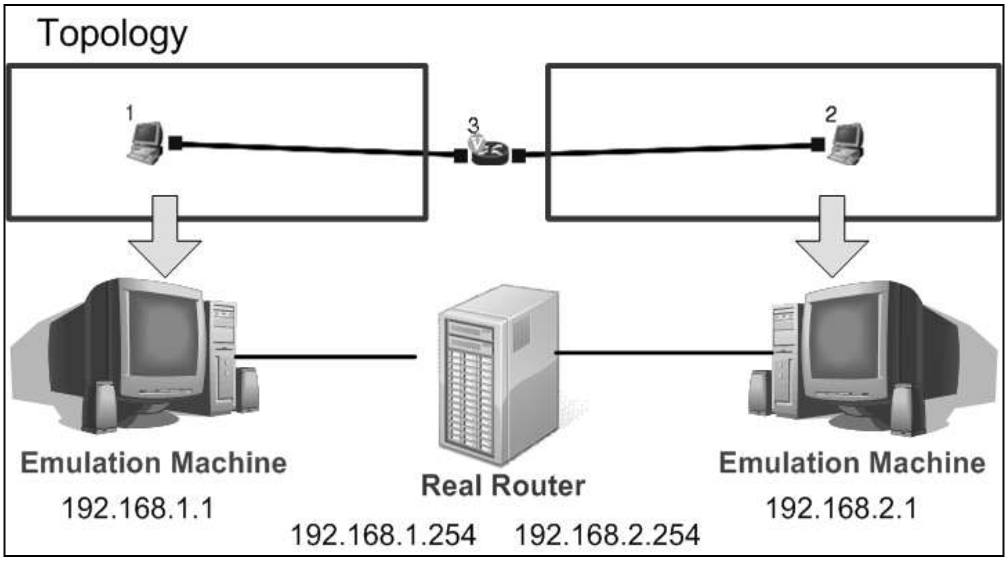


Fig 1: Real Router connected with Emulation machine

An real router is connected to two emulation machines, The IP address are shown in figure. The Dispatcher and main controller(GUI) are running on the emulated machine with IP address 192.168.1.1 , and we are running the coordinator program in emulated machine with IP address 192.168.1.2.

Steps for communication set up:

Step 1: All the above machines are physically connected to form a network

Step 2: Dispacter program is set to run on the emulated machine [192.168.1.1]

Step 3: Coordinator Program is set to run on the emulated machine [192.168.1.2], before it is set to run modify the Dispatcher IP parameter in configuration file of coordinator.

Step 4: Run the GUI on the emulated machine [192.168.1.1] and specify the IP address [192.168.1.1] to dispatcher .

Step 5: In GUI edit network topology , and run simulation.

The GUI will split in to two networks as shown I figure and give their configuration files to the corresponding emulated machine

4 Problem Faced and solutions

4.1 Details about the Problem

In this section, we discuss the Problems that occurred in communication setup of NCTUns-6.0 Network simulator.

FEDORA Version	**KERNAL version**	**NCTUns version**
Fedora-7	2.6.21.5	NCTUns-allinone-linux-2.6.21.5-f7.20070724.tar.gz
Fedora-7	2.6.21.5	NCTUns-allinone-linux-2.6.21.5-f7.20070801.tar.gz
Fedora-7	2.6.21.5	NCTUns-allinone-linux-2.6.21.5-f7.20070928.tar.gz
Fedora-7	2.6.21.5	NCTUns-allinone-linux-2.6.21.5-f7.20071030.tar.gz
Fedora-8	2.6.23.8	NCTUns-allinone-linux-2.6.23.8-f8.20071127.tar.gz
Fedora-8	2.6.23.8	NCTUns-allinone-linux-2.6.23.8-f8.20071205.tar.gz
Fedora-8	2.6.23.8	NCTUns-allinone-linux-2.6.23.8-f8.20071221.tar.gz
Fedora-8	2.6.23.8	NCTUns-allinone-linux-2.6.23.8-f8.20080213.tar.gz
Fedora-8	2.6.24.2	NCTUns-allinone-linux-2.6.24.2-f8.20080225.tar.gz
Fedora-10	2.6.27.7	NCTUns-allinone-linux-2.6.27.7-f10.20090402-2.tar.gz
Fedora-7	2.6.27.7	NCTUns-allinone-linux-2.6.27.7-f7.20090401.tar.gz
Fedora-10	2.6.27.7	NCTUns-allinone-linux-2.6.27.7-f10.20090401.tar.gz
Fedora-11	*2.6.28.9*	NCTUns-allinone-linux-2.6.28.9-f11.20090908.tar.gz
Fedora-8	*2.6.24.2*	*NCTUns-allinone-linux-2.6.24.2-f8.20080225*.tar.gz
Fedora-9	*2.6.25.9*	*NCTUns-allinone-linux-2.6.25.9-f9.20080826.tar.gz*
Fedora-12	*2.6*.31.6	*NCTUns-allinone-linux-2.6*.31.6-f12.20100113.tar.gz

We have given *Linux version development vs Year of release* in the above graph , It is seen that the current version is *Linux Kernel 2.6.33 and year of release is 2010*

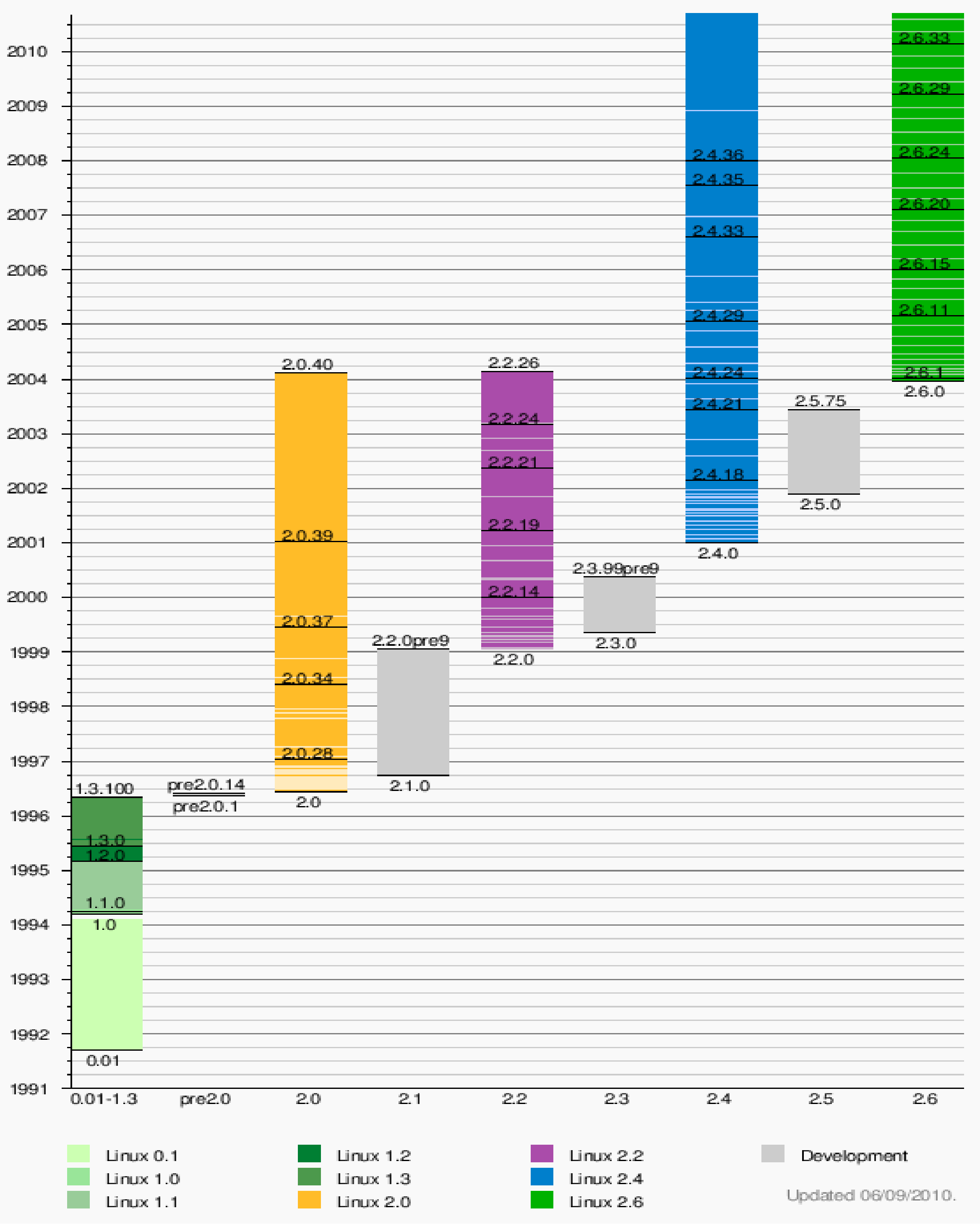
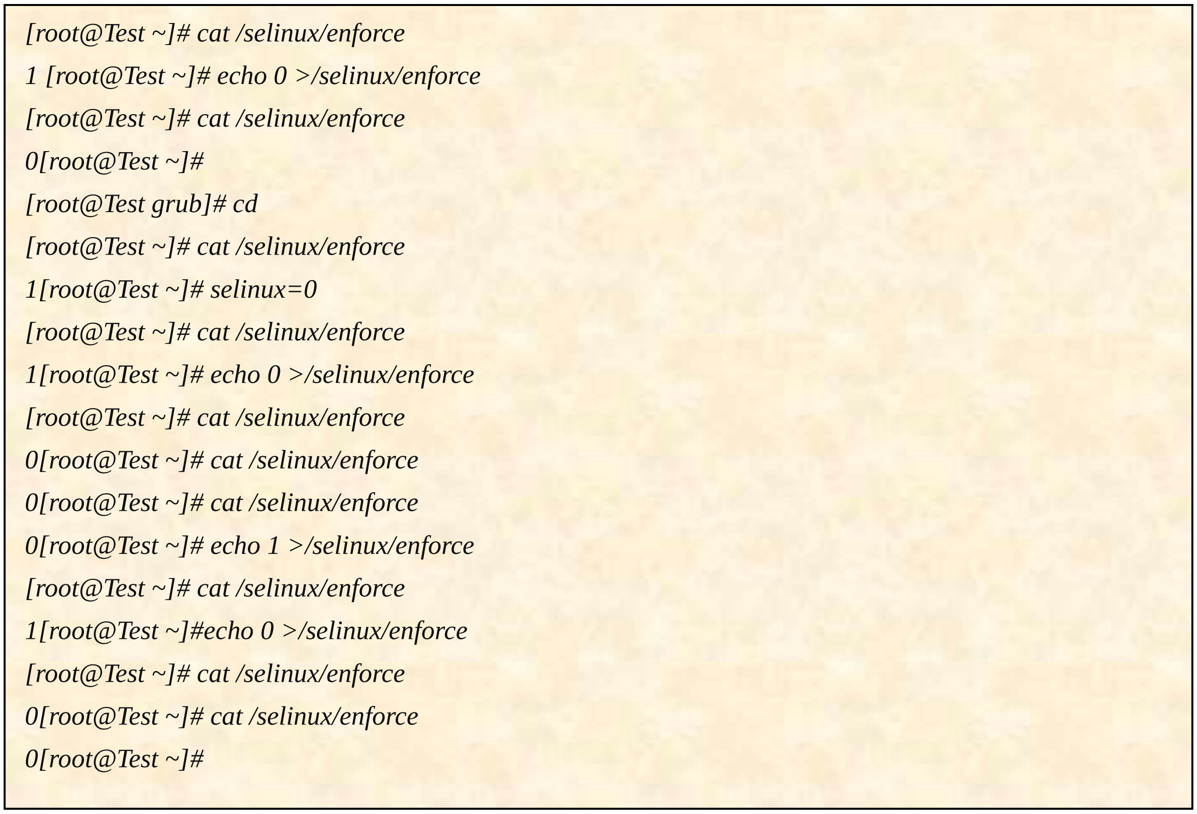


Fig 2: Linux Kernel Version (vs.) Year of release

Initially all users will follow the rules, but still the problem persists and they don’t know the basic operation and how to overcome it . They have to avoid by cross checking whether the system accepts the rules or not and also the communication is carried out in specific IP address between the coordinator and Dispatcher and GUI of NCTUns-6.0

4.1.1 Problem-1

Choose the correct version of NCTUns that matches the kernel version of Fedora Linux. We have given in table, the latest version and their kernel numbers that should be matched. There are several versions of it currently it is NCTUNS 6.0 [kernel numbers *2.6.28.9 and 2.6*.31.6, for Fedora 11 and 12]



```
[root@Test ~]# cat /selinux/enforce
1 [root@Test ~]# echo 0 >/selinux/enforce
[root@Test ~]# cat /selinux/enforce
0[root@Test ~]#
[root@Test grub]# cd
[root@Test ~]# cat /selinux/enforce
1[root@Test ~]# selinux=0
[root@Test ~]# cat /selinux/enforce
1[root@Test ~]# echo 0 >/selinux/enforce
[root@Test ~]# cat /selinux/enforce
0[root@Test ~]# cat /selinux/enforce
0[root@Test ~]# cat /selinux/enforce
0[root@Test ~]# echo 1 >/selinux/enforce
[root@Test ~]# cat /selinux/enforce
1[root@Test ~]#echo 0 >/selinux/enforce
[root@Test ~]# cat /selinux/enforce
0[root@Test ~]# cat /selinux/enforce
0[root@Test ~]#
```

4.1.2 Problem-2

Turning off SELinux temporarily

For the operating system as a whole, there is two kinds of disabling:

Permissive - switch the SELinux kernel into a mode where every operation is allowed. Operations that would be denied are allowed and a message is logged identifying that it would be denied. The mechanism that defines labels for files which are being created/changed is still active.

Disabled - SELinux is completely switched off in the kernel. This allows all operations to be permitted, and also disables the process which decides what to label files & processes with.

Disabling SELinux temporarily, if the problem you are experiencing is related to your SELinux settings. To turn it off, you will need to become the root users on your system and execute the following command:

# echo 0 > /selinux/enforce

This temporarily turns off SELinux until it is either re-enabled or the system is rebooted. To turn it back on you simply execute this command:

# echo 1 > /selinux/enforce

As you can see from these commands what you are doing is setting the file /selinux/enforce to either '1' or '0' to denote 'true' and 'false'.

Use the following procedure to disable the SELinux using the GUI.

Open the **System** menu of Fedora , By using mouse point to **Administration** and then click **Security Level and Firewall** to display the Security Level Configuration dialog box.

Then click the **SELinux** tab.

In the **SELinux Setting** select either Disabled, Enforcing or Permissive, and then click **OK**.

If you changed from Enabled to Disabled or vice versa, you need to restart the machine for the change to take effect.

Disabling SELinux could lead to problems if you want to re-enable it again later. When the system runs with file labelling disable it will create files with no label - which could cause problems if the system is booted into Enforcement mode

```
[root@Test ~]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[   OK   ]
[root@Test ~]# service iptables stop
iptables: Flushing firewall rules: [ OK   ]
iptables: Setting chains to policy ACCEPT: filter[OK]
iptables: Unloading modules:[   OK   ]
[root@Test ~]# chkconfig iptables off
[root@Test ~]# service iptables status
Table: filter
Chain INPUT (policy ACCEPT)
num   target   prot opt source                destination
Chain FORWARD (policy ACCEPT)
num   target   prot opt source                destination
Chain OUTPUT (policy ACCEPT)
num   target   prot opt source                destination
```

4.1.3 Problem-3

How Do I Disable IPtables?

First login as the root user.

Next enter the following three commands to disable firewall.

```
 # service iptables save
 # service iptables stop
 # chkconfig iptables off
```

If you are using IPv6 firewall, enter:

```
 # service ip6tables save
 # service ip6tables stop
 # chkconfig ip6tables off
```

Communication SetUp:

1. Disatcher Program is set to run:

```
[test@Test ~]$ ls
[test@Test ~]$ cd /usr/local/nctuns/bin/
[test@Test bin]$ ls
coordinator    dispatcher    nctunsclient.bin    nctunsse
[test@Test bin]$ ./dispatcher
Sorry!! dispatcher needs root's privilege to operate properly.
Please switch to root to run dispatcher
[test@Test bin]$ su root
Password:
[root@Test bin]# ./dispatcher
Environment "NCTUNSHOME" not set
[root@Test bin]# export NCTUNSHOME=/usr/local/nctuns/
[root@Test bin]# export NCTUNS_TOOLS=/usr/local/nctuns/tools
[root@Test bin]# export NCTUNS_BIN=/usr/local/nctuns/bin
[root@Test bin]# ./dispatcher
ServerSocket listen to port:9810
ServerSocket listen to port:9800
(Active:0| fd:3)      (Active:1| fd:4)
--->New Server
Server accepted: 5    2
(Active:0| fd:3)      (Active:1| fd:4)      (Active:2| fd:5)
[From Server...] register|127.0.0.1|9830|9840|IDLE
(Register Complete!)
<ServerList>:127.0.0.1:FREE
(Active:0| fd:3)      (Active:1| fd:4)      (Active:2|
```

2. Coordinator Program is set to run

```
[test@Test ~]$ su root
Password:
[root@Test test]# cd /usr/local/nctuns/bin/
[root@Test bin]# export NCTUNSHOME=/usr/local/nctuns/
[root@Test bin]# export NCTUNS_BIN=/usr/local/nctuns/bin
[root@Test bin]# export NCTUNS_TOOLS=/usr/local/nctuns/tools
[root@Test bin]# ./coordinator
/usr/local/nctuns//bin/
ServerSocket listen to port:9830      FD:4
ServerSocket listen to port:9840      FD:5
ServerSocket listen to port:9880      FD:6
UnixDomainSocket Bind Path:/tmp/nctuns      FD:7
[To Dispatcher...]    register|127.0.0.1|9830|9840|IDLE
[From Dispatcher...] OK
```

3. nctunsclient program is set to run

```
[nctuns@Test ~]$ su root
Password:
[root@Test nctuns]# cd /usr/local/nctuns/bin/
[root@Test bin]# export NCTUNSHOME=/usr/local/nctuns
[root@Test bin]# export NCTUNS_BIN=/usr/local/nctuns/bin
[root@Test bin]# export NCTUNS_TOOLS=/usr/local/nctuns/tools
[root@Test bin]# ls
coordinator   nctunsclient        nctunsse       printPtr       sat.sim   sat.xtpl
dispatcher    nctunsclient.bin   nctunsxterm   sat.results   sat.tpl
[root@Test bin]# ./nctunsclient
mkdir /root/.nctuns
mkdir /root/.nctuns/etc
mkdir /root/.nctuns/tmp
Session management error: None of the authentication protocols specified are supported
mkdir /usr/local/nctuns/bin/satdemo1.sim
mkdir /usr/local/nctuns/bin/satdemo1.results
[root@Test bin]#
```

5. Conclusions and Future Works

Thus Distributed Emulation can be carried out between real world applications with a simulator. Thus NCTUns-6.0 is a simulator based on kernel re-entering and distributed Emulation simulation methodology, with this methodology a more realistic network model can be simulated for real time applications. Here Coordinator is a Program that sets up communication between the GUI and Dispatcher in NCTUns-6.0 I, a network simulator running on Fedora-13. This paper presents problems faced in communication set up of Coordinator with GUI and Dispatcher . This paper focuses on how to overcome the above problems and execute simulations in real time. This paper also deals with KDE, SELINUX, iptables, environment variables and X-server for NCTUns-6.0 simulation engine

References

Jungkeun Yoon, Mingyan Liu, and Brian Noble, "Random Waypoint Considered Harmful", *IEEE INFOCOM*, IEEE 2003.

Michael Feeley, Norman Hutchinson, and Suprio Ray, "Realistic Mobility for Mobile Ad Hoc Network Simulation",*ADHOC-NOW 2004*, pp.324–329, Springer-Verlag Berlin Heidelberg, 2004.

Mirco Musolesi, and Cecilia Mascolo, "A Community Based Mobility Model for Ad Hoc Network Research", *REALMAN'06*, May 26, 2006, Florence, Italy.

M. Musolesi, S. Hailes, and C. Mascolo. "An Ad Hoc Mobility Model Founded on Social Network Theory". *In Proceedings of MSWiM'04*, pages 20–24. ACM Press, October 2004.

A. Jardosh, E. Royer, K. Almeroth, and S. Suri. "Towards realistic mobility models for mobile ad hoc networks." *In MobiCom'03*, September 2003.

X. Hong, M. Gerla, G. Pei, and C.-C. Chiang. "A Group Mobility Model for Ad hoc Wireless Networks". *In Proceedings of the ACM/IEEE MSWIM'99, Seattle, WA*, August 1999.

James M. Cargal, "Social relationship as Algebraic Relationship", *Proceedings of the 1980 Winter Smulation Conference*, copyright IEEE 1980.

C. Bettstetter, G. Resta, and P. Santi. "The node distribution of the random waypoint mobility model for wireless ad hoc networks", *IEEE Transactions on Mobile Computing,* 2(3):257–269, July-September 2003.

Wensi Xi, Benyu Zhang, and Zheng Chen, "Sim Fusion : Measuring Similarity using Unified Relationship Matrix ", *SIGIR'05,* August 15–19, 2005, ACM Press ,2005.

S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A distance routing effect algorithm for mobility (DREAM)". *In ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom98)*, pages 76-84, 1998

Mohsen jamali and Hassan Abolhassani, "Different aspects of socials network analysis". *In IEEE/WIC/ACM International Conference on web intelligence, (WI'06),* 2006.

X. Zeng, R. Bagrodia, and M. Gerla. "Glomosim: A library for parallel simulation of large-scale wireless networks". *In Proceeding of PADS Workshop*, May 1998.

C. E. Perkins and E. M. Royer, "*The Ad hoc on-Demand Distance Vector Protocol. In C"* E. Perkins, editor, Ad hoc Networking, pages 173–219. Addison-Wesley, 2000.

B. Karp, "Geographic Routing for Wireless Networks", *PhD thesis*, *Harvard University*, 2000.

M Sanchez. Mobility models. http://www. disca.upv.es/misan/mobmodel.htm. Page accessed on May 30th, 2007.

http://pcl.cs.ucla.edu/projects/glomosim

OMNeTpp. 2007. Available online via <http://www.omnetpp.org/> [accessed Feb. 15, 2006]

Authors

D.SatishKumar received his B.E., degree in Electrical and Electronics Engineering from Bharathiar University and M.E. degree in Computer Science and Engineering from Manonmaniam Sundharnar University respectively. He is currently working towards the PhD degree in Computer Science and Engineering in Anna University of Technology Coimbatore, Tamil Nadu INDIA. His research interests accumulate in the area of IEEE 802.16j Multi hop Mobile networks, Routing, WIMAX networks, Network management in Wireless Networks.

Email: - satishcoimbatore@yahoo.co.in

Dr.N.Nagarajan received his B.Tech and M.E. degrees in Electronics Engineering at M.I.T Chennai. He received his PhD in Faculty of information and communication engineering from Anna University, Chennai. He is currently working as Principal, Coimbatore Institute of Engineering and Technology, Coimbatore. He is member of board of study of faculty of information Technology at Anna University of technology, Coimbatore. His specialization includes optical, wireless Adhoc and sensor networks. He is guiding assorted research scholars in optical networks and wireless networks.

Email: - swekalnag@rediffmail.com