IOP Conf. Series: Materials Science and Engineering **434** (2018) 012273 doi:10.1088/1757-899X/434/1/012273

Teaching network security in Linux using Netkit with implementation virtual laboratory

Y Ariyanto*, B Harijanto and Y W Syaifudin

Information Technology Department, State Polytechnic of Malang, East Java, Indonesia

*yuri.bjn@gmail.com

Abstract. Security is important in computer networks. Network security is the task of the Administrator, where the administrator is responsible for data security on the network. The problem that is often faced by a network administrator is the laboratory as a test site of network design that has been created without interrupting the running system. Testing laboratories are needed by the administrator as a test site for network design, before implementing the design on a real network system. In this research, the researchers propose a method to solve the problem of testing the computer network, through the approach of using virtual machine by creating virtual laboratory. This virtual laboratory allows an administrator to quickly reset the configuration without interrupting the system that has been running. The virtual laboratory is based on UML (User Mode Linux) virtualization technology using Netkit emulators. The researchers verified the experiment through a simulation based on firewall network security design trial scenarios that have been created using the Netkit emulator. The results of trials in a virtual laboratory show the similarity of the working environment to a real network system. It can be concluded that the Netkit emulator is able to run test scenarios and firewall commands on network security properly.

1. Introduction

A network emulator is a software in which the environment can mimic the original functions and habits of the network [1]. Network emulators are used by administrators, researchers, teachers for the implementation of real network design into network emulators, making it easier to test scenarios that have been created [1]. In network learning, network emulator can be use as test environments.

Virtualization tools have been used in computer network classes to provide a environment for practice experiments. Although some papers describe these tools and their technical benefits, they do not provide quantitative information about its effectiveness [2].

In this paper describe network administrators often face problems in conducting trial scenarios of network designs created. Where in the test should not interfere with the system that has been running. Therefore, network administrators need testing labs to implement the network scenarios that have been created.

In this research, researchers proposed a method to solve the problem of computer network testing, through the approach of using a virtual machine by creating a virtual lab. This virtual lab allows administrators to quickly reset configurations without interrupting the running system. This virtual laboratory is based on UML (User Mode Linux) virtualization technology using Netkit emulators. For testing, it is done by testing the network security design scenario using linux iptables firewall.

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

2. Literature review

2.1. Network emulator

A network emulator is a software in its environment mimicking the functions and habits of the original network [1].

2.2. User Mode Linux (UML)

User Mode Linux (UML) is a Linux virtual machine running on Linux. Technically, UML is a port connection from Linux to Linux. Linux has connected ports to many different processors, including X86, Sun's SPARC, IBM and Motorola's PowerPC, DEC's (Compaq and HP) Alpha and various other processors [3].

UML has been widely used for network system administrators, network system developers and users. UML is different from other virtualization technologies because it is a Virtual Operating System (OS), but to call UML can be done virtually. UML technologies such as VMWare are truly virtual machines by copying physical platforms, from the CPU to the device, even though the running OS on the physical platform also runs on the emulator platform provided by VMWare. In VMWare any OS that runs on a platform can be booted under VMWare, otherwise UML can only be a guest OS on Linux [4].

2.3. Netkit

Netkit is the place to set up and conduct network experiments at low cost and with little effort. The Netkit opensource network simulator was created by a group of university professors who use it as a tool in their teaching. It allows to create multiple virtual network devices (routers, switches, computers, etc.), which can be easily connected to form a network on one PC. The network equipment is virtual but has many original characteristics including the configuration interface [4-8].

2.4. Linux iptables

The Linux kernel's network packet processing subsystem is called Netfilter, and iptables is the command used to configure it. Iptables is the software firewall that is included with most Linux distributions by default [9-10].

The iptables architecture groups network packet processing rules into tables by function (packet filtering, network address translation, and other packet mangling), each of which have chains (sequences) of processing rules. The rules consist of matches (used to determine which packets the rule will apply to) and targets (that determine what will be done with the matching packets). Iptables operates at OSI Layer 3 (Network). For OSI Layer 2 (Link), there are other technologies such as ebtables (Ethernet Bridge Tables) [9,11].

3. System design

3.1. Hardware and software specification

At this stage, the needs of computer hardware specifications used to design and implement virtual laboratories using Netkit will be explained. Hardware and software requirements are as follows:

- OS Debian 7
- RAM 2 GB, Hardisk 320 GB
- Processor Intel Dual Core
- Emulator Netkit (Netkit-2.8.tar.bz2, Netkit filesystemi386-F.2.tar.bz2, Netkit-kernel-i386-K2.8.tar.bz2)

3.2. Network topology

Virtual lab experiments implementing iptables firewall security is depicted on the network topology implemented on Netkit, as shown in Figure 1.

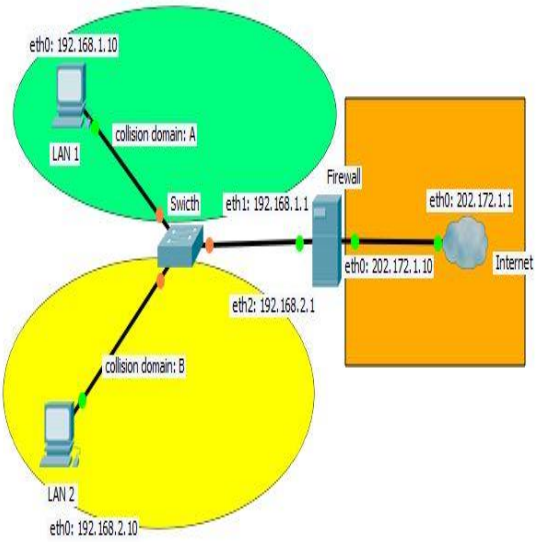


Figure 1. Network topology.

The description of Lab network topology is as follows. Firewall is shown in table 1.

Table 1. Design details.

No	Computer Name	Lan Card	Ip Address
1	LAN1	eth0	192.168.1.10/24
2	LAN2	eth0	192.168.2.10/24
3	Firewall	eth0	202.172.1.10/24
	Firewall	eth1	192.168.1.1/24
	Firewall	eth2	192.168.2.1/24
4	Internet	eth0	202.172.1.1/24

4. Lab scenario

At this stage, the lab will be described. The virtual lab. firewall is created using netkit emulator. Virtual labs are implemented on a network basis the network topology design in Figure 1. The testing scenario of virtual laboratory implementation is shown in the following stages.

4.1. Create a file lab.conf

Create a lab.conf file is done to set the network scenario according to the network topology design the virtual lab. firewall, in accordance with the configuration syntax of the netkit emulator. The implementation of the lab.conf file is shown in Figure 2.

```
File  Edit  View  Search  Terminal  Help
 GNU nano 2.2.6                File: lab.conf

firewall[0]=tap,202.172.1.1,202.172.1.10
firewall[1]=A
firewall[2]=B

lan1[0]=A
lan2[0]=B
```

Figure 2. lab.conf.

In lab.conf to connect connections between NICs every computer is virtually used collosion domain. Collosion domains are symbols that connect virtually between NICs on a Netkit emulator.

Implementation of the virtual lab. firewall consists of a virtual machine as a firewall computer, lan1 and lan2.

4.2. Configure Virtual Machine (VM) in lab.conf

At this stage researchers will configure the .startup file on Netkit. This file is required for the configuration and service to run when the virtual machine is started.

4.2.1. Configuring VM firewall. Create a firewall.startup file is show in Figure 3.

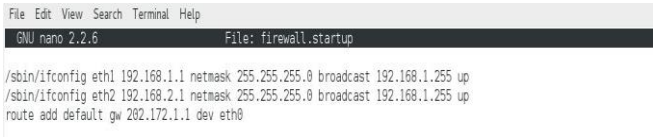


Figure 3. A screenshot of firewall.startup.

File and configuration for virtual machine are shown in Figure 3, with ip address eth1 192.168.1.1/24, eth2 192.168.2.1/24 and add ip gateway 202.172.1.1 therefore the firewall virtual machine can be connected to the internet virtual machine.

4.2.2. Configuring VM lan1. Create a lan1.startup file is shown in Figure 4.



Figure 4. A screenshot of lan1.startup.

File and configuration for virtual machine are shown in Figure 4, with ip address eth0 192.168.1.10/24 and add ip gateway 192.168.1.1 therefore the lan1 virtual machine can be connected between another virtual machine.

4.2.3. Configuring VM lan2. Create a lan2.startup file is shown in Figure 5.

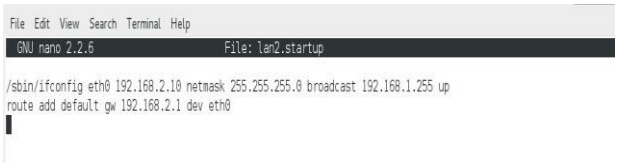


Figure 5. A screenshot of lan2.startup.

File and configuration for virtual machine are shown in Figure 5, with ip address eth0 192.168.2.10/24 and add ip gateway 192.168.2.1 so that the lan2 virtual machine can be connected between another virtual machine.

5. Labs implementation

Implementation of the network topology design Figure 1. on the Netkit emulator is shown in Figure 6.



Figure 6. Virtual Lab firewall in Netkit emulator.

Results of virtual lab implementation will be described as follows. Firewall is shown in Figure 6. In the lab virtual machine ip address is configured. Iptables Rule used in virtual lab. Firewall is shown in Figure 7.

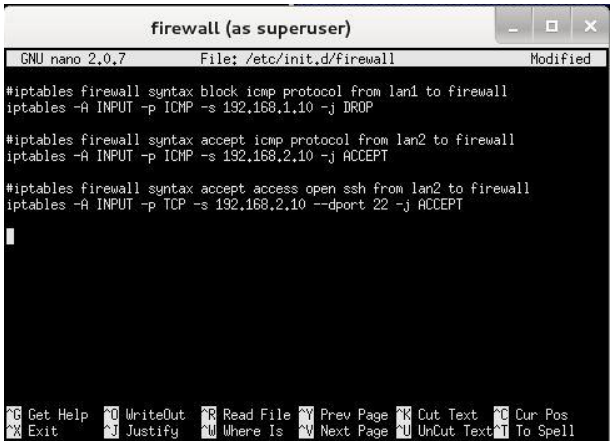


Figure 7. Rule iptables in VM firewall.

6. Testing and analysis virtual lab firewall

This stage will be tested firewall rules that will run on the virtual machine firewall. Figure 8 shows how iptables firewall rules are run on a virtual machine firewall, with the command linux #/etc/init.d/ firewall start and #iptables –L.

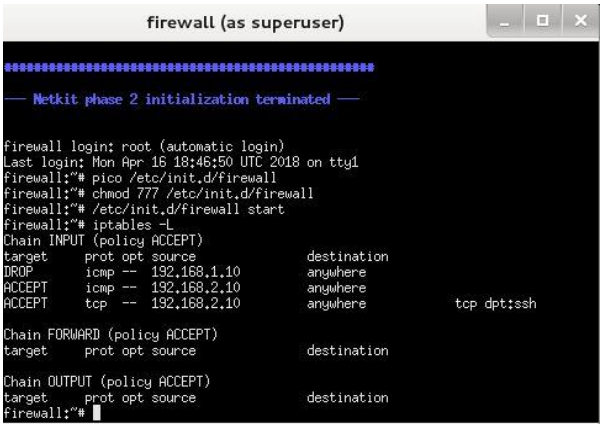


Figure 8. Rule iptables firewall running.

6.1. Testing virtual lab. firewall

Figure 9 shows the ICMP block test protocol, pinging from the lan1 virtual machine to the firewall virtual machine.

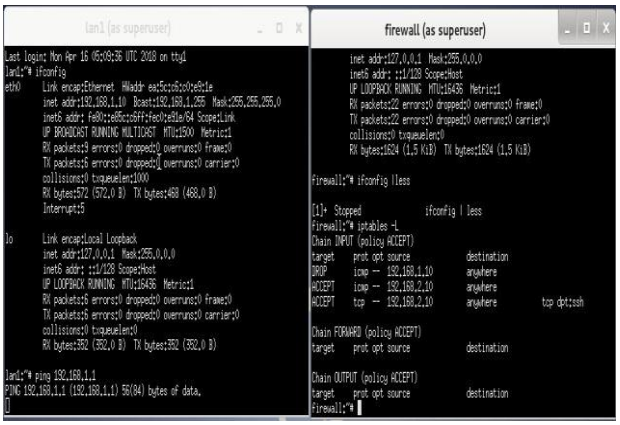


Figure 9. Blocking ICMP protocol with iptables.

Figure 10 shows an ICMP Protocol accepted, with a ping command from lan2 virtual machine to firewall virtual machine.

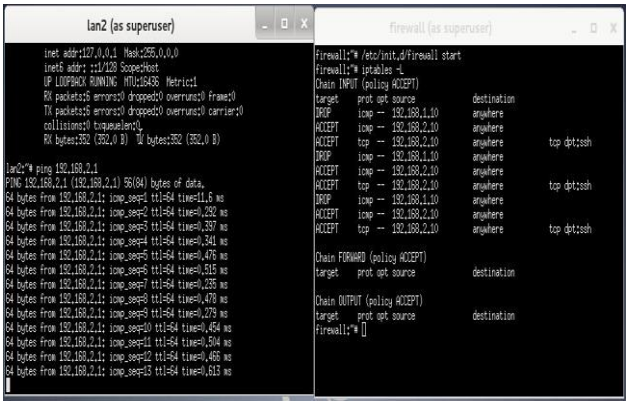


Figure 10. Accepted ICMP protocol with iptables.

Figure 11 shows a remote access SSH with open port 22 from lan2 virtual machine to firewall virtual machine.

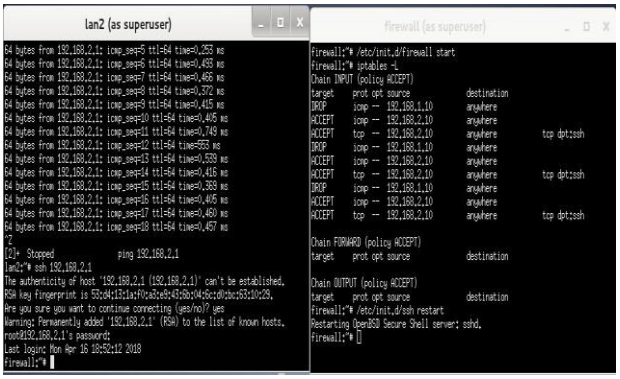


Figure 11. Open port 22 SSH.

6.2. Analysis virtual lab firewall

Analysis of virtual machine experiment in virtual lab. Firewall is shown in table 2.

Table 2. Analysis of iptables commands.

No	VM 1	VM 2	Rule Iptables	Status
1	lan1	Firewall	Drop ICMP Protocol	Running
2	lan2	Firewall	Accept ICMP Protocol	Running
3	lan2	Firewall	Open port 22 SSH	Running

7. Conclusions

From the test results, it can be concluded, all iptables firewall commands can be run in virtual lab. firewall in Netkit, the results of trials in a virtual laboratory show the similarity of the working environment to a real network system and it can be concluded that the Netkit emulator is able to run test scenarios and firewall commands on network security properly

References

[1] Ariyanto Y, Syaifudin Y W and Harijanto B 2017 Performance analysis of network emulator based on the use of resources in virtual laboratory *Electrical Engineering, Computer Science and Informatics (EECSI), 2017 4th International Conference* IEEE p 1-6

[2] Gurgel P H, Branco L H, Barbosa E F and Branco K R 2013 Development of a practical computer network course through Netkit virtualization tool *Procedia Computer Science* **18** p 2583-2586

[3] Dike J 2006 *User Mode Linux* (Prentice Hall)

[4] University of Roma Tre Computer Networks Research Group 2018 *Netkit http://www.netkit.org*. accessed on 2 April 2018

[5] Fermin G, David F, Javier R, Omar W and Tomas de Miguel 2004 Use of Virtualization Tools in Computer Network Laboratories *Proc. 5th International Conference on Information Technology Based Higher Education and Training (ITHET 2004)* p 209-214

[6] Maurizio Pizzonia and Massimo Rimondini 2008 Easy Emulation of Complex Networks on Inexpensive Hardware *Proc. 4th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities (TRIDENTCOM 2008)* Innsbruck

[7] Massimo Rimondini 2007 *Emulation of Computer Networks with Netkit* (Technical Report RT-DIA-113-2007, Roma Tre University)

[8] Hung Nguyen, Matthew Roughan, Simon Knight, Nick Falkner, Olaf Maennel and Randy Bush 2010 How to Build Complex, Large-Scale Emulated Networks *Proc. 6th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities*

(TRIDENTCOM 2010) Berlin
[9] Gregor N. Purdy 2004 *Linux Iptables Pocket Refences* O'Reilly.*www.oreilly.com*
[10] Qing-Xiu Wu 2013 The Research and Application of Firewall based on Netfilter. 2012 International Conference on Solid State Devices and Materials Science *Procedia Computer Science* **18** p 2583 – 2586
[11] Larry Peterson and Bruce Davie. *Computer networks - A systems approach, 3rd Edition* (Morgan Kauffman)