

Flexible Cloud Environment for Network Studies

Vindya Wijeratne

School of Electronic Engineering
and Computer Science, Queen
Mary, University of London
Mile End Road
London E1 4NS

vindya.wijeratne@eecs.qmul.ac.uk

Joel Obstfeld

Juniper Networks (UK) Ltd.
Building 1 Aviator Park
Station Road, Addlestone
Surrey KT15 2PG

jobstfeld@juniper.net

Chris Phillips

School of Electronic Engineering
and Computer Science, Queen
Mary, University of London
Mile End Road
London E1 4NS

chris.phillips@eecs.qmul.ac.uk

ABSTRACT

This paper describes a novel cloud-based virtualization platform that provides an advanced teaching and research environment for network studies. The arrangement allows for the rapid and automated configuration of various network scenarios. The cloud architecture is explained and a typical use-case scenario is described where the cloud facility provides an undergraduate / postgraduate introduction to routing within a realistic environment. The approach is compared and contrasted with an earlier teaching facility that employed an array of rack-mounted computers running Quagga routing software. The assessment demonstrates that not only is the virtualization platform a practical and efficient alternative providing an “anywhere, anytime” capability with consistent performance relative to typical distributed network emulation platforms, it is also able to accurately emulate the behaviour of commercial routing systems.

I. INTRODUCTION

This paper outlines a joint project between Juniper Networks and Queen Mary, University of London (QMUL), involving the use of hundreds of Virtual Machines running the Juniper Network Operating System (JUNOS) hosted on a cloud computing platform located in Sunnyvale, California. It provides a valuable opportunity to augment the theoretic principles that students learn in the classroom by both configuring and observing routing protocol operation in a captive network environment. In this project, 240 VMs were employed for the study of the Routing Information Protocol (RIP) and Open Shortest Path First (OSPF) routing protocols by Masters students undertaking a module on Internet Infrastructure and undergraduate students on the Internet Protocols module. This amounted to over two hundred students, some of whom were remotely located in Portugal, Spain, China and in the UK.

The experiments involved examining a number of routing-protocol configuration parameters and carrying out tests such as exploring the consequences of link failures that could be seen in real-world scenarios.

1.1 Background

In the past, similar experiments were undertaken using a physical testbed located at QMUL that consisted of Linux-based PCs running Zebra [1] or Quagga [2] routing software as shown in Figure 1.

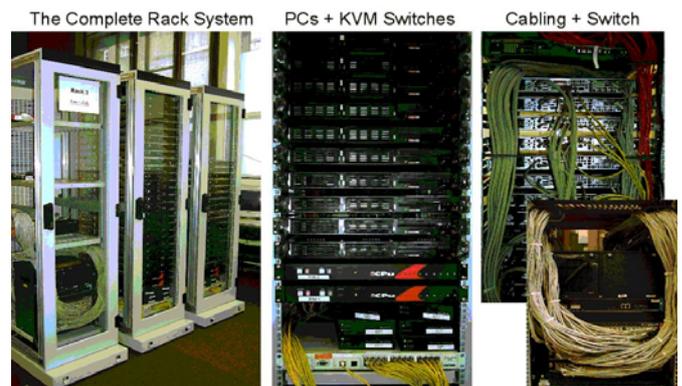


Figure 1: Photograph of Previous QMUL Testbed

Each of the PCs had 5 network interfaces; one of which was connected via a bridge to a server and functioned as the control/management segment. The other four interfaces were connected to a router/switch, which provided in-band network connectivity. This arrangement, together with the use of Norton’s Ghost software, enabled the rapid configuration the PCs via the control/management segment using a number of pre-configured images. An imaging cycle would take about 30 seconds to complete. In addition, the router/switch could be used to reconfigure the network interconnections by arranging interfaces into the appropriate VLAN groups. The overall architecture is summarised in Figure 2 and further details are given in [3].

Typically the PCs were arranged into clusters of 8 machines for each student workgroup. In addition, packet-sniffing PCs and a breakout patch panel were added to provide further flexibility. The patch-panel, for instance, allowed the students to physically unplug cables to produce link failures in order to observe the reconvergence behaviour of different routing protocols. The PCs were also connected via KVM switches to a reduced number of terminal stations. This negated the need for air conditioning within the laboratory and saved considerable space.

Despite the validity of this approach, which was employed for over 6 years, there were a number of

drawbacks. The main ones were the logistics of providing well over one hundred students with access to the teaching facility in a relatively short space of time, as demanded by the constraints of the course. This resulted in high student to equipment ratios. It also required a dedicated room to be set aside to house the equipment. Furthermore, in later years, physical stresses on the equipment took their toll and a number of components such as network interface cards had to be replaced, though this was aggravated on one occasion by a lightning strike that caused power supply problems.

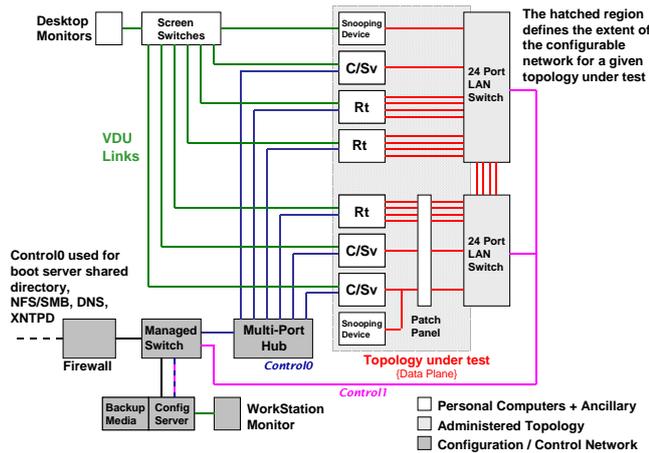


Figure 2: Previous QMUL Testbed Architecture

1.2 Alternative Virtual Lab Environments

The challenge faced at Queen Mary is by no means unique and to that end, a number of approaches have been adopted to create environments in which networks can be modelled. Many have used either ‘user process’ routing or switching entities or have used a form of a virtual-machine, and come in a range of size and scope. Whilst some of these solutions are executed on a local computing platform, others which require more computing resources in order to model larger networks or host larger numbers of network entities, make use of distributed virtual machine instances hosted on remotely-located computing resources provided by other groups. For example, Emulab [4] provides one such emulated network testbed which gives researchers a wide-range of environments to conduct experiments; however, users need to write their own routing applications or use an existing application such as Quagga which isn’t as realistic as experimenting with an actual commercial router platform. Planetlab [5] and Seattle testbed [6] operate on resources donated by various institutions, and the latter provides a broader resource diversity though the computers are not dedicated to the platform. Further, due to the nature of this architecture, users do not have control over the underlying hardware which may cause problems with respect to repeatability of experiments. Netkit [7] is another popular network testbed which comes at low-cost; however, the

scalability is limited. Also, Netkit uses open-source software based on User Mode Linux.

A number of these solutions make use of Open-source routing daemons such as Zebra or Quagga and XORP [8]. However, the range of features available with open-source software usually tends to be limited in comparison with that of a commercial router implementation. The value of the distributed approach is clear in that the scale that can be achieved (in many cases) is significantly larger than that which a single educational entity could reasonably afford if they were to use physical computing devices rather than ‘virtualised’ devices. The remotely hosted approach is not that dissimilar from the ‘Cloud Computing’ model which has developed over the past few years except that most Cloud environments are owned by a single entity who is able to manage the computing resources as a whole. Having a single management entity can have operational advantages (e.g., platform and OS consistency, troubleshooting) which are much more difficult to achieve when there are many owners in a distributed computing environment.

The Juniper Networks platform extends the advantages of Cloud Computing and provides a solution that meets the teaching and research needs of QMUL. With a ‘Platform as a Service’ delivery method, the Juniper Networks environment provides the students with invaluable practical exposure to a commercial Network Operating System that they may encounter in real-world networks, while also avoiding many of the drawbacks of previous approaches. The rest of the paper is organised as follows. Section 2 explains the Juniper Networks cloud architecture and Section 3 briefly describes the experiments undertaken by the students. Section 4 discusses the numerous benefits of employing this cloud-based virtualization platform and section 5 presents some possible future directions.

2. JUNIPER CLOUD PLATFORM

The Juniper Networks solution is a cloud-based virtualisation platform where multiple VMs running a range of operating systems can be configured to create network topologies. While the primary operating system for VMs configured to act as routers is JUNOS, other virtual machines can be configured as Unix devices (using Linux or FreeBSD) or Customer Premise Equipment (using OpenWRT). Furthermore, a VM can be configured to act as a network tool such as a Traffic-Generator (using Spirent TestCenter Virtual) or a Routing Analytics engine (using Juniper Networks’s Route Insight Manager). Support for Unix-distributions provides an environment in which widely-used network tools such as Multi Router Traffic Grapher (MRTG) or tools developed by the user can be deployed.

2.1 Architecture

As shown in Figure 3, the virtualization platform makes use of the Linux KVM hypervisor as its virtualisation layer,

together with QEMU to provide the required machine models for the different virtual machine Guest-OS's. High-density rack-mounted x86 servers, typical for a data-centre, are used in a 'loose-cluster' configuration, with the VM operation, configuration, placement and network topology being managed by a Juniper Networks proprietary Virtual Machine Manager (VMM) environment.

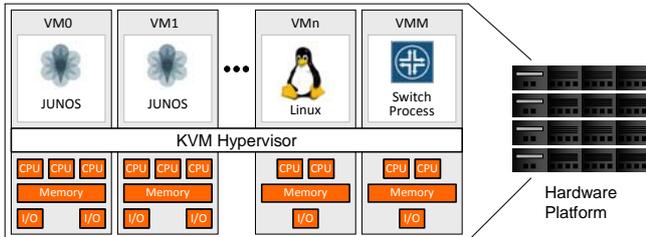


Figure 3:- Basic Cloud Structure

The VMM system parses configuration files that describe the set of virtual machines required, the operating system to be loaded on each VM, the connectivity between the VMs as well as applying per-VM configuration settings (if necessary).

Console-access to an emulated serial port as well as an out-of-band management Ethernet port is provided to each VM so that users operate the virtual machine in exactly the same manner as they would if they had access to a physical device.

A Juniper Networks proprietary solution enables the creation of Layer-2 Ethernet connectivity between the in-band network segments present in the virtualised network topology and physical Ethernet segments in the physical world. A secure tunnel is established between the physical location and the virtualization platform, over which Layer-2 frames are passed, enabling physical devices such as routers, to communicate and exchange data-plane and control-plane frames to virtual routers as if they were co-located on the same Ethernet segment.

2.2 Juniper Networks Virtual Router Instances

The VMs running the JUNOS Network Operating System are referred to as Virtual Routers (VRs). The VR implements a software-based forwarding plane, enabling the use of advanced networking features such as Multicast, MPLS Traffic Engineering Tunnels, Virtual Private LAN Services, MPLS-based VPN services (including Carrier-supporting-carrier) in addition to the full range of control-plane and management-plane features (such as SNMP and XML) present in the JUNOS operating system when used on physical routers.

Since the virtual router is running the full JUNOS operating system rather than a partial system emulation, the VR is operated and managed in exactly the same way a physical router, exposing the user to the same Command Line Interface command structure and output as well as configuration syntax that would be employed [9].

3. ROUTING EXPERIMENTS

The experiments carried out by students on the Internet Infrastructure module were based on each student workgroup having 8 VRs running RIP and OSPF, interconnected as illustrated by Figure 4.

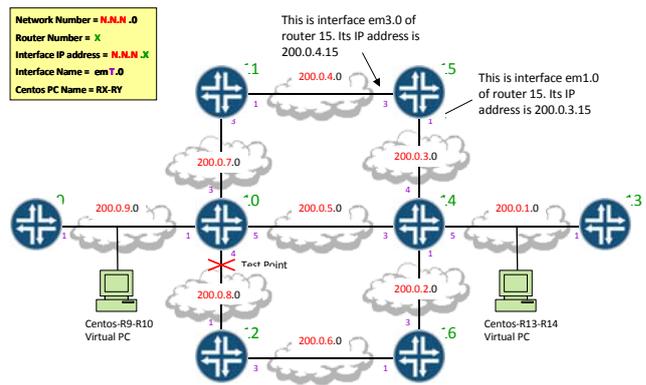


Figure 4 – Test scenario

The experiments consisted of six parts as described below.

- Part 1: Routing Information Protocol (RIP) Basics – all 8 VRs were pre-configured to run RIP version-2 and in this part students examined the contents of routing tables and ping/traceroute results by executing relevant JUNOS operational mode commands and tcpdump captures (e.g. to identify various header fields and the time interval between RIP messages) facilitated by the CENTOS PCs attached to the network.
- Part 2: Examining Failure and Reconvergence with RIP – in this part, students were asked to mimic a link failure at the 'Test point' shown in Figure 4 which was realised by disabling interface-4 of VR-10. Hence, students were given the opportunity to work on the JUNOS configuration mode. The paths taken before and after the link failure were compared, whilst also noting the time taken to reconverge (which was determined based on the sequence numbers of the ping messages).
- Part 3: Open Shortest Path First (OSPF) Basics – this part provided further experience on the configuration mode by allowing students to configure all 8 VRs to run OSPF instead of RIP. Once configured, students

made various OSPF-related observations such as OSPF databases, neighbours, Hello interval, cost etc.

- Part 4: Examining Failure and Reconvergence with OSPF – similar to Part 2, a ‘link failure’ was introduced and the paths taken before and after the breakage were compared. Tcpcmdump was also started in order to capture any link state messages exchanged. Moreover, the reconvergence time was compared with that of Part 2 and students were asked to discuss any differences in the speed of reconvergence between the two protocols.
- Part 5: OSPF Parameter Negotiation and Cost Setting – in this part, while all VRs were still configured with OSPF, students learnt the function of the Hello interval with regards to forming adjacencies by varying the value of the Hello interval and examining the status of OSPF neighbours. The students also examined the effect of modifying the cost metric.
- Part 6: RIP / OSPF Route Redistribution – in this part, some of the VRs remained configured for OSPF while some others were configured for RIP. One VR was configured for route re-distribution between RIP and OSPF by defining an additional policy. The students then examined the routes taken to various destinations and compared and analysed why some of these routes differ to the RIP-only or OSPF-only cases, and why the path to two interfaces of the same VR can sometimes be markedly different under this configuration.

As shown in Figure 4, a single test scenario requires 8 VRs and 2 virtual PCs, i.e. a total of 10 VMs. 240 VMs were made available to approximately 75 students, enabling workgroups of 3 students to have their own unique set of VMs. Each student was provided with their own Radius-authenticated account along with the logging of command history.

As described above, the students were encouraged to question and analyse the observations gathered. They were also asked to write a report including their analyses, together with an essay on improving the reconvergence time of RIP and OSPF. Excellent pieces of work have been produced for which we believe gaining first-hand experience on commercial equipment has been a major contributing factor.

4. DISCUSSION

Practical labs reinforce concepts described in the lecture room. They also allow the students to hone their investigative skills. In this instance, students capture realistic data using applications such as Tcpcmdump and have to assimilate the information to determine whether the system is performing as expected or whether there are faults.

In the latter case, the student has to decide what further actions are necessary to isolate the fault and to resolve it.

The Juniper Networks cloud platform has enhanced the students’ educational experience by enabling the completion of a complete learning cycle as described in Kolb’s experiential learning theory, i.e. concrete experience, reflection, abstract conceptualisation and active experimentation [10]. The students were able to experience the application of taught material on real equipment, reflect on those observations, analyse and draw conclusions which could then be re-applied in gaining hands-on experience for further reflection and so on. This type of repeated learning cycles (known as a continuous spiral) was made possible as the platform was available for an extended period of time, giving students scope to experiment. This has provided a deeper learning experience (rather than superficial learning) as encouraged in pedagogical science and is indeed of significant value for network engineers who need to adapt to changing environments.

In addition to students based at QMUL, this platform was used by Distance Learning (DL) students located throughout the world, e.g. in Europe and Macao. Since access was restricted to QMUL IP addresses, the DL students accessed the platform via a VPN connection to QMUL. Even via the VPN, the round trip times experienced were still reasonable and all students have been able to carry out the experiments as if they were sitting in front of a physical router.

Despite being complete beginners with respect to knowledge of Juniper Networks’s JUNOS operating system (and the majority being unfamiliar with any network OS), the students were seen to handle the experiments very well. Documentation and online interactive training provided by Juniper Networks have certainly helped [9]. Experimental results could easily be saved onto text files and transferred to local machines via SCP. Tasks such as returning to the base configuration could be easily done at any time by executing simple commands.

In essence, this virtual platform offers a number of advantages over the physical testbed previously used at QMUL. For instance, the virtual platform has great flexibility, e.g. in changing the topology and examining link failures anywhere on the network. In the previous setup, the link failure was only readily supportable at a pre-defined location where the patch-panel provided students with access to a single point of failure. The virtual environment also provides greater scalability than was achievable with the physical resource present at QMUL. The cloud environment permits the virtualisation of hundreds of virtual router instances, all accessible via a telnet or SSH connection from a terminal that can be located anywhere. This allows for the point of delivery to be any generic computing facility within the educational establishment. The cloud can also be quickly setup to replicate a base

configuration many times. In the scenario described in Section 3, this meant that the topological configuration of 8 virtual routers and 2 virtual PCs was only created once. This configuration can then serve as a template that can be replicated as many times as needed. This approach also serves to provide a library of configurations that can be shared across a user community, promoting reusability and allowing novice experimenters to exploit existing “tried and tested” scenarios or to build upon them to quickly create personalised adaptations to suit the topic under consideration. The application was proven by the reuse of the QMUL lab exercises and teaching material by groups at Princeton, Boston, Loughborough and Vienna universities.

Perhaps the greatest benefit of the virtualization environment is the fact that the virtual routers are complete emulations of their physical namesakes. In terms of their control-plane behaviour, including the routing messages created and how the routers are setup, they are identical to their real world counterparts. This effectively gives students access to an unprecedented quantity of networking equipment that would be hitherto prohibitively expensive. It also enables students to get hands-on experience of working with these routing devices in a practical manner and to develop the fault diagnosis skills that will serve them well in a real situation.

In order to provide a richer environment, the virtualization platform supports not only the use of Virtual Machines loaded with operating systems such as Linux or FreeBSD but virtual-machine-based applications. Figure 5 gives an example where Juniper Networks’s Route Insight Manager (RIM) is shown in use. RIM is a routing analytics engine that is able to process routing updates from common Interior Gateway Protocols (IGPs) and Border Gateway Protocol (BGP) in order to provide a real-time view of the network. In Figure 5, a RIM Virtual Machine has been connected to a virtualised network composed of JUNOS Virtual Routers. Providing such tools helps the students more clearly understand the environment in which they are working since they are able to ‘see’ the network. Furthermore, it provides an introduction to the kinds of network management tools that are used in real-world networks.

There are obviously drawbacks when using a virtualised environment. For example, data-plane behaviour of the VRs does not accurately emulate its physical equivalent – a Gigabit Ethernet interface in a virtualised environment does not necessarily have the same throughput, latency, jitter etc. of such an interface in the physical world. This means that studies focussing on the traffic behaviour are not ‘realistic’. Packet scheduling characteristics and the virtual link speeds are not a fair reflection of the behaviour that would be observed from real devices. In this respect, the only solution remains a ‘captive office’ containing actual hardware or a carefully crafted simulation environment where the

simulation models have been setup to appropriately mimic the scheduling and forwarding behaviour that would be experienced by the data flows. Similarly, care must be taken when considering aspects of network convergence, since time in a virtualised network is not the same as that in the physical world as far as the virtual machines are concerned.

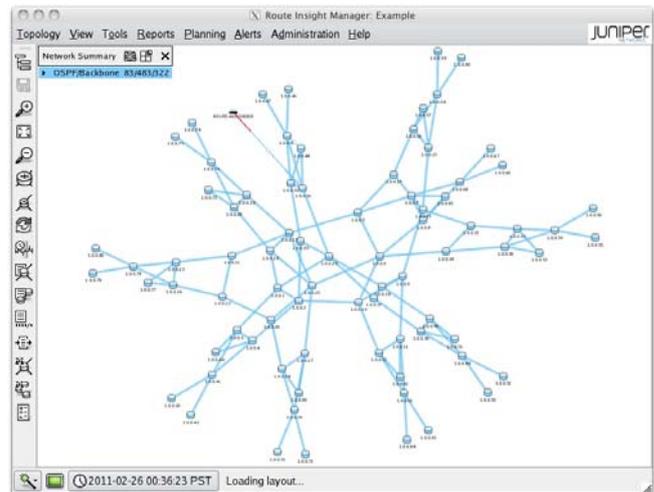


Figure 5 – RIM visualizing in a virtual network

5. CONCLUSIONS AND FURTHER WORK

The joint project described in this paper has been a great opportunity for students in gaining hands-on experience of cutting-edge real-world technologies and demonstrating the relevance of the Internet related programmes that are offered. Exposure to relevant industrial equipment within a university environment is highly beneficial for students. Formal and informal student feedback has shown that this has not only been a valuable experience, but also an enjoyable one. Moreover, a number of students are now looking to undertake their MSc thesis on themes employing this virtual platform. In addition to what was done in the described experiments, the platform is also capable of supporting IGPs such as IS-IS and Exterior Gateway Protocols such as BGP. This opens doors for further projects especially by research students and staff and relevant discussions are currently in progress.

6. ACKNOWLEDGEMENTS

The authors would like to thank the following institutions for their participation in the program:

Princeton University – Prof. Jennifer Rexford
 Boston University – Prof. David Starobinski
 Loughborough University – Dr. Olaf Maennel
 University of Vienna – Prof. Kurt Tutschku
 Juniper Networks Education Services – Nicolas Xenos

7. REFERENCES

- [1] GNU Zebra, Free routing software distributed under General Public License (GNU). Retrieved 20th February 2011 from <http://www.zebra.org/>
- [2] Quagga Routing Software Suite. Retrieved 20th February 2011 from <http://www.quagga.net/>
- [3] Phillips, C. , Marzo, J.L. et al. Flexible Distributed Testbed for High Performance Network Evaluation, *2nd International IEEE/Create-Net Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM, Barcelona, Spain, March 1-3, 2006)*, Page(s):567 – 573.
- [4] Emulab: Network Emulation Testbed. Retrieved 26th February 2011 from <http://www.emulab.net/>
- [5] Planetlab: An open platform for developing, deploying and accessing planetary-scale services. Retrieved 26th February 2011 from <http://www.planet-lab.org/>
- [6] Seattle: Open Peer-to-Peer Computing. Retrieved 26th February 2011 from <https://seattle.cs.washington.edu/html/>
- [7] Netkit: The poor man's system to experiment computer networking. Retrieved 26th February 2011 from http://wiki.netkit.org/index.php/Main_Page
- [8] XORP: eXtensible Open Router Platform. Retrieved 26th February 2011 from <http://www.xorp.org/>
- [9] Juniper Networks Technical Documentation. Retrieved 03rd March 2011 from http://www.juniper.net/techpubs/en_US/release-independent/junos/information-products/pathway-pages/junos/product/index.html
- [10] Certificate in Learning and Teaching (CILT) course notes, provided by The Learning Institute, Queen Mary, University of London, 2008.