Cong Li

Kernel-based Virtual Machine

Bachelor's thesis
Information Technology

2017

South-Eastern Finland
University of Applied Sciences

South-Eastern Finland University of Applied Sciences

Author (authors)	Degree	Time
Cong Li	Bachelor of Information Technology	May 2017
Title Kernel-based Virtual Machine		49 pages 0 pages of appendices
Commissioned by		
Supervisor Matti Juutilainen		

Abstract

The purpose of this project was to introduce about the basic virtualization technology and make people know how to manage the KVM virtual machine and what they could do with the KVM virtual machine.

The methods I mainly used in this project were the KVM remote management, bridged network, KVM live migration, DRDB Pacemaker HA cluster and etc.

The result of the work was that bridged network was created, and I could manage the KVM virtual machines remotely with some remote tools and KVM live migration allowed virtual machines to be migrated between different hosts and DRBD Pacemaker HA cluster provided a higher level of availability, reliability, and scalability for clients.

Keywords

operating system, virtualization, KVM, bridged network, HA cluster

CONTENTS

1 INTRODUCTION

The thesis subject I selected is one kind of virtualization product- Kernel-based Virtual Machine. Before starting with this topic, KVM was totally a new thing for me. However, I have still some virtualization study experience to talk about. First, I have some Linux skills that I can make directories, install some applications, add users and so on. Additionally, I have explored one sort of similar Virtual Machine-VMware, with which I set up my own web server, like Nginx or Apache web server, with which I constructed LAMP (Linux, Apache, MySQL, PHP) environment with which I did some labs related to Windows server and Linux server during university's studies.

After discussing the topic with my supervisor, he gave me many precious and useful suggestions about the KVM subject. Meanwhile, I searched for some material about each topic that my supervisor suggested. I will cover the main topics as bellowing, like explaining what is virtualization, Linux operating system installation, KVM installation, network related configuration, how to manage or monitor KVM hosts remotely, how to migrate VMs and making the KVM hosts work in Cluster. The next chapter will introduce more detail about KVM subject.

2 THEORETICAL PART

Virtualization technology is becoming more and more popular in the world, and increasing number of famous companies have started to develop their own virtualization technology. People could get all kind of different virtualization products according to their demands and preferences nowadays. According to Kenneth (2010), VMware, Oracle, Red Hat, KVM, OpenVZ and more virtualization products are becoming similar to users in the market. Microsoft yielded with Hyper-V, to compete with VMware for the server virtualization market. Red Hat purchased Qumranet in 2008 and released the SPICE protocol as open source in 2009. Additionally, like Oracle, Amazon, and Google and so on, all of them own different core virtualization products or technology. All the companies producing virtualization products are attempting to improve and develop the virtualization products to occupy the virtualization market.

In this theoretical part, I will introduce the virtualization environment and compare some different virtualization application, such as VMware, Hyper-V, KVM at first. Then, I will concentrate on KVM in detail. The goal of this part is to make people have distinct understanding of virtualization and KVM.

2.1 Linux Operating System

The operating system is a piece of software that manages all the hardware and all the other software. For instance, it controls every file, device, section of main memory, and nanosecond of processing time. At the same time, the operating system provides an interface to the user and the other software. Without the operating system, the software won't function. There are five essential managers in every operating system and they are processor manager, memory manager, device manger, file manager and network manager. (Tutorialspoint 2017.)

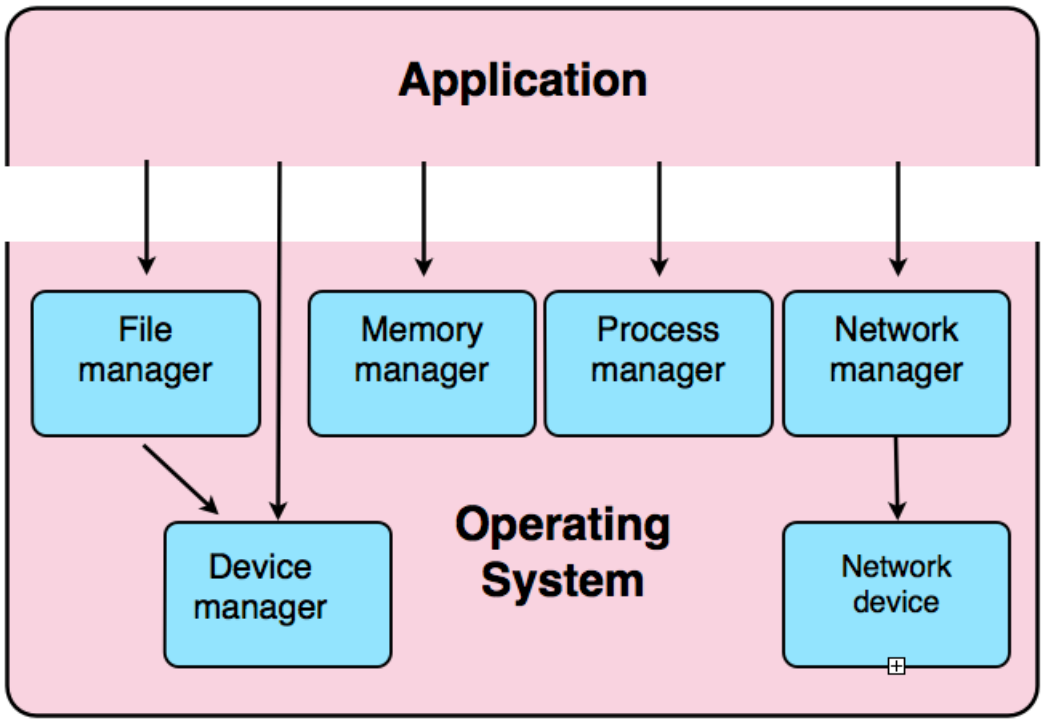

Figure 1. Essential managers in operating system

There are various operating systems in the world, such as Linux, Windows, iOS, etc. I will select one kind of Linux distribution operating system, the Ubuntu

desktop operating system to install in my computer. According to Wikipedia (2017), the Linux is a Unix-like operating system, designed to provide personal computer users a free or very low-cost operating system. Linux's kernel was developed by Linus Torvalds at the University of Helsinki in Finland. According to Prabesh (2013), comparing with Windows operating system, Table 1. below lists reasons why many people prefer to choose Linux operating system as below:

Table 1. Advantages of the Linux operating system

• Open source	It means that customers are free to make any change that they like in the operating system. And many people can contribute to the code.
• Easy to install applications	Installing new applications in Ubuntu is easier than in Windows. Because its all open source, You don't need to accept agreements then click a few times "Next" before the program is installed.
• Security	Open source code allows users to search for the viruses and malicious attacks easier, then to fix them.
• Free of cost	A lot of Linux operating distributions are allowed to be download from the correspondingly official website.

Ubuntu is Debian-based Linux operating system for computers, smartphones, servers, etc. On the desktop, Ubuntu has been widely spread by small and large enterprises due to its simple management, security model, various management tools and free charging. Currently the two most popular Ubuntu desktop operating systems versions are Ubuntu 16.04 LTS (Long Term Support) and Ubuntu 16.10. Ubuntu offers thousands of apps available for download. Most are available for free and it can be installed in a simple way. (Canonical Ltd. Ubuntu 2017.)

I download and install the last Ubuntu desktop operating system version, Ubuntu 16.04 LTS. According to Canonical Ltd. Ubuntu (2017), there are five essential requirements for installing Ubuntu 16.04 LTS, including, 2 GHz dual core

processor, 2 GB system memory, 25 GB of free hard drive space and a DVD drive or a USB port for the installer media.

2.2 Virtualization

According to Margaret (2016), virtualization describes a technology in which an application, guest operating system or data storage is abstracted away from the true underlying hardware or software. Virtualization technology is widely used in server virtualization which uses a software layer called a hypervisor to emulate the underlying hardware. Common virtualization approaches involve network virtualization, application virtualization, desktop virtualization, storage virtualization and server virtualization. Virtualization technology is widely used in the world now. According to Juutilainen (2017), Table 2 I listes some reasons why virtualization is significant and what kinds of benefits virtualization could bring to us. (See Table 2).

Table 2. Advantages of virtualization technology

• Simple management	Traditionally, we only run one operating system on one server. With virtualization we can put multiple virtual machines onto a single physical server. This would be much simple for management.
• Better energy efficiency	Multiple guest operating systems are working on a single physical server that reduces the number of physical servers as well as reduces the requirements for cooling system and battery backups.
• Testing environment provision	Virtualization provides software developers with a better environment to do a variety of testing.
• High availability	A virtual machine is not tied to the physical server. If the physical server

	goes down, it is free risk to install the virtual machine to another physical server.

2.3 Vmware

VMware Inc. was founded in 1998 and is a subsidiary of Dell Technologies. VMware provides multiple software services, including cloud and virtualization software and services. Like the other virtual machines, VMware allows more than one operating systems running simultaneously in one single physical workstation or server. VMware's desktop software is running on Microsoft Windows, Linux, and macOS, while its enterprise software hypervisors are designed for servers. VMware ESX and VMware ESXi are both bare-metal hypervisors that are applied on server hardware without requiring an additional underlying operating system. (Wikipedia 2017.)

VMware software enables complete hardware to the guest operating system, such as video adapter, network interface card, hard disk and I/O devices. In practice, administrators have the options that pausing operations on a virtual machine guest, moving or copying that virtual machine to another physical computer, and the suspension function allows the administrator to resume the execution exactly from the point of suspension. In terms of enterprise servers, vMotion feature allows the migration of operational guest virtual machines between separate hardware hosts shared nothing. VMware becomes one of the most popular virtualization product in the world, and it dominates the server virtualization market and desktop virtualization market. (Wikipedia 2017.)

VMware becomes one of the most popular virtualization products in the world, and it dominates the server virtualization market and desktop virtualization market.

2.4 Hyper-V

As the virtualization technology has developed, Microsoft must take into account the virtualization trend. Hyper-V emerged in 2008 as a virtualization platform, and it keeps upgrading new Hyper-V versions. So far, four versions of Hyper-V has been released, which are Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2 and Windows Server 2008. (Microsoft 2017.)

Hyper-V is a role in Windows Server 2008 and Windows server 2008 R2. Hyper-V provides a platform where users can build a virtualized server computing environment. Hyper-V is significant because people can build and manage multiple virtual machines on one physical computer, and server resources can be more efficiently used as well. (Microsoft 2017.)

The three most essential requirements of Hyper-V are needed. Firstly, an x64-based processor, Hyper-V is to be used in x64-based versions of Windows Server 2008, Windows Server 2008 Standard, Windows Server 2008 Enterprise, and Windows Server 2008 Datacenter. Secondly, hardware-assisted virtualization is required, like Intel Virtualization Technology (Intel VT) or AMD Virtualization (AMD-V). The third one is data execution prevention (DEP). DEP must be enabled, especially the Intel XD bit or AMD NX bit must be enabled before setting up Hyper-V role.

2.5 KVM

KVM is short for Kernel-based Virtual Machine, and currently it has developed one of the most popular Virtual Machines in the world. According to Red Hat, Inc. (2016), KVM is a kind of complete virtualization which is based on computers' hardware support, like Intel VT and AMD-V. According to Wikipedia (2017), KVM has to be also migrated into FreeBSD and illumos in the form of loadable kernel modules. KVM is applied to various guest operating systems, including Linux, Windows, Solaris, Haiku, ReactOS and so on.

Based on Scale Computing (2017), I found some advantages of KVM, and I will list them as follows:

1. Support for any kind of guest OS
2. Efficient code
3. Open source and flexible
4. No licensing costs

3 KVM VIRTUALIZATION ENVIRONMENT

KVM is a full virtualization for Linux on the x86 hardware virtualization extensions (Intel VT or AMD-V). The loadable kernel module, kvm.ko, is the core virtualization part of KVM and it provides the core virtualization infrastructure and a processor specific module, kvm-intel.ko or kvm-amd.ko. (Red Hat, Inc. 2016.)

With KVM, multiple virtual machines can run under the unmodified Linux or Windows images. Every single virtual machine has its own virtualized hardware, containing network interface card (NIC), storage devices, graphical user interface (GUI), etc. (Red Hat, Inc. 2016.)

Compared with VMware and Hyper-V, KVM is open source software. The kernel component of KVM, KVM kernel module was merged into Linux 2.6.20 and has since been fixed as a part of the kernel. (Red Hat, Inc. 2016.)

3.1 KVM Prerequisites

Before installing KVM, we need some prerequisites. According to Ubuntu documentation (2016), installation requires a processor, which can be AMD or Intel or whatever that supports hardware virtualization to run KVM. Intel and AMD both have developed extensions for their processors, deemed respectively Intel VT-x and AMD-V. Meanwhile, I will explain what is Intel VT-x and AMD-V respectively.

3.1.1 Intel VT

Intel VT is Inter virtualization technology, which is a platform where processors are running virtualization. Intel VT has been divided into three extensions for computer hardware virtualization, including Inter VT-x, VT-d, VT-c. Different Intel VT extension has its own advantages. Here is a table below showing the different features of each Intel VT extension. (Margaret 2009).

Table 3. Intel virtualization technology

• VT-x	VT-x adds migration, priority and memory handling capabilities to a wide range of Intel processors.
• VT-d	VT-d adds virtualization support to Intel chipsets that can assign specific I/O devices to specific virtual machines (VM)s.
• VT-c	Comparing with VT-d, VT-c brings better virtualization support to I/O devices such as network switches.

3.1.2 AMD-V

AMD-V is AMD virtualization, which is a set of hardware extensions for the X86 processor architecture. AMD Virtualization technology was first announced in 2004 and added to AMD's Pacifica 64-bit x86 processor designs. By 2006, AMD's Athlon 64 X2 and Athlon 64 FX processors appeared with AMD-V technology, and today, the technol-ogy is available on Turion 64 X2, second- and third-generation Opteron, Phenom and Phenom II processors. Advanced Micro Dynamics designed the extensions to perform repetitive tasks normally performed by software and improve resource use and virtual machine performance. (Margaret 2009).

3.2 KVM Hypervisor

There are two different hypervisor types. Type 1 hypervisor which is installed directly on the host hardware. Type 2 hypervisor is installed as a software application on an operating system. According to Margaret (2017), KVM hypervisor is a piece of program in Kernel-based Virtual Machine (KVM). A

hypervisor provides multiple operating systems to share computer resources in a single host. In KVM, the Linux kernel plays a role as a Type 2 hypervisor. The hypervisor builds VM environments and provides coordination calls for hardware resources like, processor, memory, hard disk, network, etc. over the host OS.

3.3 QEMU

QEMU is a monitor of virtual machine in the host: it emulates CPUs through dynamic binary translation and provides a set of device models, enabling it to run a variety of unmodified guest operating systems. It can also be used together with KVM in order to run virtual machines at near-native speed (requiring hardware virtualization extensions on x86 machines). QEMU can also do CPU emulation for user-level processes, allowing applications compiled for one architecture to run on another. (Wikipedia 2017.)

3.4 Libvirtd

According to Wikipedia (2017), libvirt is an open source API, daemon and management tool for managing platform virtualization. It can be used to manage KVM, Xen, VMware ESX, QEMU and other virtualization technologies. These APIs are widely used in the orchestration layer of hypervisors in the development of a cloud-based solution. (Wikipedia 2017.)

libvirt itself is a C library, but it has bindings in other languages, notably in Python, Perl, OCaml, Ruby, Java, JavaScript (via Node.js) and PHP. libvirt for these programming languages is composed of wrappers around another class/package called libvirtmod. libvirtmod's implementation is closely associated with its counterpart in C/C++ in syntax and functionality. (Wikipedia 2017.)

3.5 KVM Network

According to Ubuntu KVM/Networking (2016), there is a few different ways to allow virtual machines to access to the external network services. The default virtual network configuration is known as Usermode Networking, but for this

networking part, I will concentrate on introduction about Bridged Networking. Next chapter will introduce it more in detail.

3.5.1 Bridged Networking

Accordint to Jamie (2015), A bridged network shares a real Ethernet device with virtual machines (VMs). Each VM can bind directly to any available addresses, either IPv4 or IPv6 on the LAN. According to Ubuntu KVM/Networking (2016), bridged networking let the virtual interfaces communicate with the external network through the physical interface of the host. Virtual machines act as real physical computers connecting with other hosts over the internet. The greatest benefit of bridged networking is that it allows virtual machines to interact with other virtual machines or physical hosts.

Bridged networking does not support, as the physical network devices used for bridging are wireless devices. There are two different ways to bridge the network. First one, in commonly server, without GUI, you need to install the bridge-utils package, use ***brctl*** and the ***/etc/network/interfaces*** file to set up bridges. Another method is applied- Networking Manager. (Ubuntu documentation 2016). According to Dan (2015), the GUI tool: ***nm-connection-editor*** is the simplest way to create bridging network with Networking Manager. In order to avoid any configuration issues, It is a good idea to choose the higher version of Network Manager.

3.6 VirtManager

VirtManager provides users an graphical user interface to manage the virtual machines. It is widely used in KVM virtual machines, but as well as used to manages Xen and LXC (linux containers). It records a summary information of virtual machines, like running state, or resource utilization etc. Additionally, an embedded VNC (virtual net-work computing) client viewer would be provided to display a full graphical console to the guest virtual machines. (Daniel P. 2013.)

Virtmanager uses the libvirt virtualization library to manage the available hypervisors. libvirt exposes an application programming interface (API), which is integrated with a large number of open source hypervisors, to enable control and monitoring. Virtmanager supports the mangement tools including, like virt-install tool. Virsh-install is a command line tool and it provides an easy way to provision operating systems into virtual machines. virt-clone tool is a command line tool as well and which is applied for cloning inactive virtual machines. It copies the disk images, and assigns a config with a new name, UUID and MAC address pointing to the copied disks. virt-viewer application is an interface used to interact with the graphical display of virtualized guest OS. It can present VNC or SPICE, and uses libvirt to scope for the graphical connection details. (Jones 2012.)

3.6.1 OpenSSH

OpenSSH is short for Open Secure Shell. It is a free suite of tools which help to make network communications more security by encrypting the data of network traffic. OpenSSH provides an effectively elimination of all kinds of network level attacking, such as eavesdropping, connection hijacking and so on. (OpenBSD 2017.)

OpenSSH started as a fork of the free SSH program, developed by Tatu Ylönen; later versions of Ylönen's SSH were proprietary software, offered by SSH Communications Security.OpenSSH was first released as part of the OpenBSD operating system in 1999. (Wikipedia 2017.)

OpenSSH is not a single computer program, but rather a suite of programs that serve as alternatives to unencrypted network communication protocols like FTP and rlogin. Active development primarily takes place within the OpenBSD source tree. OpenSSH is integrated into the base system of several other BSD projects, while the portable version is available as a package in other Unix-like systems. (Wikipedia 2017.)

3.7 Migration

According to Techopedia Inc. (2017), virtual machine migration is a process of moving virtual machine from one host to another physical host. Users must shut down the machines during the offline migration, however, live migration provides user without powering off the machines during the migration. According to Red Hat, Inc. (2017), KVM currently supports savevm/loadvm and offline or live migration. Users must shut down the machines during the offline migration, however, live migration provides user without powering off the machines during the migration.

Migration commands are given when in qemu-monitor. Upon successful completion, the migrated VM continues to run on the destination host. Migration allows users to migrate virtual machines between Intel hosts and AMD hosts. A 32-bit virtual machine can be migrated to both a 32-bit and 64-bit host, but a 64-bit hosts only supports to migrate to a 64-bit host. (Red Hat, Inc. 2017.)

3.7.1 Basic Requirements

According to Fedora documentation (2017), live migration requirements as follows:

1. The same version of KVM must be run on the virtual machines.
2. The virtual machine image must be located on a shared storage (eg, using iSCSI, NFS, GFS2 or Fiber Channel) and the shared storage must be accessible on both source and destination host.
3. Both source and destination host must be on the same subnet.
4. TCP protocols must open on source host and destination host.
5. OpenSSH or other remote applications should be installed on both source host and destination host.

3.7.2 Network File System (NFS)

According to Margaret (2016), Network File System was originally developed by Sun Microsystems in the 1980's and is now managed by the Internet Engineering

Task Force (IETF). According to Ubuntu documentation (2017), the network file system (NFS) is a client application which allows hosts to share, update files or directories over the network remotely. Some benefits to use the Network File System as bellow:

- Data accessed by all users can be stored on a central host, with clients mounting this directory at boot time. For instance, you can keep all user accounts on one host, and have all hosts on your network mount /home from that host. With NFS, users can then log into any system, and still work on one fix of files. (Andrew 1996.)

- Less disk space is used for the local workstation, because the shared data can be stored in the single host which provides accessible to others over the network. (Ubuntu documentation 2017.)

- To make things much simple. Users do not have to make separate Home direc-tories and install the same file on every single machine, instead, users need only to build Home directories on the NFS server and shared it over the network. (Ubuntu documentation 2017.)

3.7.3 Iptables

According to Korbin (2014). Iptables is a command-line firewall, built for Linux operating systems. According to Ubuntu documentation (2017), Iptables is applied to all the Ubuntu distributions, like Ubuntu, Xubuntu, Kubuntu by default. There is a lot of basic Iptables options, which involes –A, -L, -m, -p and etc. Iptables uses policy chains to allow or block traffic. Iptables comes into three different chains: Input, Forward, and Output.

Input – This chain is applied to manage the incoming connections. For instance, if a user wants to allow TCP into your Ubuntu host, iptables will match the host' IP address and TCP port to a rule in the input chain. (Korbin 2014.)

Forward – Administrators can manage where packets should be routed within a LAN with Forward policy. (Red Hat, Inc. 2017.)

Output – This chain is used for outgoing connections. For instance, if a user tries to ping some website, like www.google.com, iptables will check its output chain to view what the rules are regarding ping and www.google.com before making a decision to deny or allow the connection attempt. (Korbin 2014.)

3.7.4 KVM/Virsh

Virsh is a tool that allows user to manage the virtual machines, like creating, deleting, running, stopping from the command line. Virsh is widely used for advanced Linux administrators who are interested in script or automating some aspects of managing their virtual machines. (Ubuntu documentation 2014.)

3.8 Introduction To Server Cluster

According to Microsoft (2003). A server cluster is a group of independent servers (more than two servers) working together to provide a higher level of availability, reliability, and scalability for clients. According to INDUSTRY NEWS, WEB HOSTING (2016). There are the reasons why we select the server cluster as follows: (See Table 4).

Table 4. The benefits of server cluster

Highly availability	Server cluster is designed to provide high availability solution. In case, one server is down and another server from the cluster will take over the server having problems and countinue to work.
Scalability	Resources can be added to nodes accordingly and ench node from the server cluster would share the resources.
Server maintenance	If one server from the cluster requires maintenance, it can be stopped and other servers will take over its job and

	the clustered servers can be configured and managed anytime without any trouble.

3.8.1 Cluster Basics

According to Red Hat. Inc. (2017), there are four major types of clusters which are storage cluster, high availability, load balancing and high performance. (See table 5).

Table 5. Four major types of clusters

Storage cluster	Storage clusters provide a consistent file system image across servers in a cluster, allowing the servers to simultaneously read and write to a single shared file system.
High availability cluster	High-availability clusters provide continuous availability of services by eliminating single points of failure and by failing over services from one cluster node to another in case a node becomes inoperative.
Load balancing cluster	Load-balancing clusters dispatch network service requests to multiple cluster nodes to balance the request load among the cluster nodes.
High performance cluster	High-performance clusters use cluster nodes to perform concurrent calculations. A high-performance cluster allows applications to work in parallel, therefore enhancing the performance of the applications.

3.8.2 High Availability Cluster (HA cluster)

A high availability cluster consists of two or more computers that act like a single system and provide continuous uptime. And a high availability cluster allows virtual machines (VMs) on a given host to fail over to another host without any downtime in case of a failure.The purposes of the high availability cluster are to provide load balancing, backup and failover. (Margaret 2013.)

Although high availabitity clusters can range from two nodes to numbers of nodes, administrator must think about the number of VMs and hosts he or she adds to an HA cluster because too many VMs and hosts can complicate load balancing. (Margaret 2013.)

3.8.3 DRDB

According to Ubuntu documentation (2017), DRDB is short for Distributed Replicated Block Device, which is mirrors block devices between two or more computers. According to THE URBAN PENGUIN (2016). DRDB manages replication of the data at the block device level. Any block devices, like hard disks, partitions, RAID devices, logical volumes and so on can be mirrored. The replication is completely transparent to other applications on the host systems. The most different and nice thing is that DRBD builds a shared-nothing cluster and we do not need the central shared SAN, iSCSI and etc.

4 PRACTICAL PART

In this practical part, I will carry out some tests based on the theory part. The practical part includes series the process of Ubuntu desktop operating system and KVM installation, a serial of configuration related to KVM, like network configuration, hardware configuration and so on. Then I will show the skills like how to make the KVM hosts work in cluster, and how to manage KVM hosts and migrate VMs between them. Generally speaking, I will specify what I did and how I did it my practical part.

4.1 Ubuntu Operating System Installation

Before installing the Ubuntu operating system, there are two essential things that I need to do. First, I downloaded Ubuntu desktop operating system version- Ubuntu 16.04.1 LTS iso from the Ubuntu official website to my USB flash drive. Then I installed this Ubuntu operating system to my external SSD. The list below describes the process of how I installed it gradually.

1. Using a USB flash drive- I plugged my USB flash drive into the computer and powered on my computer. Then I pressed F9 to get access to the boot menu and chose the boot device- SSD.

2. Preparing to install- I selected Download updates while installing and installed this third- part software.

3. Installation type- For this step, I would choose any installation type what I like. I chose the type ‘Something else’.

4. Doing partitioning- My SSD’s space is 120 GB, I split 80 GB for Root, 10 GB for Swap and 30 GB for /home.

5. Location and preferred keyboard layout selection- I selected location in Finland and use the US keyboard as my preferred keyboard.

6. Login password setting- I just choose one username and correspondingly password. Then I just click ‘continue’ button.

7. Restart my computer and enjoy the Ubuntu. That’s it.

4.2 Installing KVM

KVM only works if the CPU has hardware virtualization support- either Intel VT-x or AMD-V. Based on Chris (2012), I understand how to Install KVM and create virtual machines on Ubuntu as follows.

Before installing KVM, I need to test whether my CPU processor includes these features. With the command:

egrep -c '(svm|vmx)' /proc/cpuinfo

The number 0 explains that your computer processor does not support hardware virtualization, while the number 1 or more indicates that your computer does. From my test I got the number 8 which tells me that my computer supports the hardware virtualization. (see Figure 2).

Figure 2. Hardware virtualization test

Now, I start to install KVM and supporting packages: Virt-Manager and libvirt. The command as follows:

sudo apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager

After the installation finished, if I are not the root user, I should add the users in the libvirtd group with running the following command to add your user account to the libvirtd group: sudo adduser name libvirtd. I added user “cong” and added “cong” to the libvirtd group.

Then, by typing the command:

virsh -c qemu:///system list

I should get an empty list of virtual machines, which indicates that everything is working fine. The picture should be like in Figure 3:

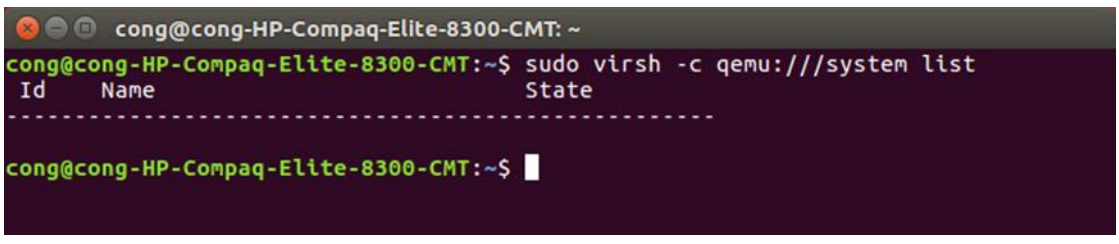

Figure 3. KVM & Libvirt installation

4.3 Network Related Configured

I will use Networking Manager to configure bridged networking. Based on Dan (2015), here is the process how to create bridged networking gradually, and how the ***nm-connection-editor*** looks like and works.

Firstly, I created two virtual machines with the Ubuntu guest operating system in my Ub-untu host. One was named Ubuntu_Guest, the other was Ubuntu_Test. Then I start configur-ing a Linux bridge with Network Manager on Ubuntu via ***nm-connection-editor***. The picture below shows how the ***nm-connection-editor*** looks like. (see Figure 4).

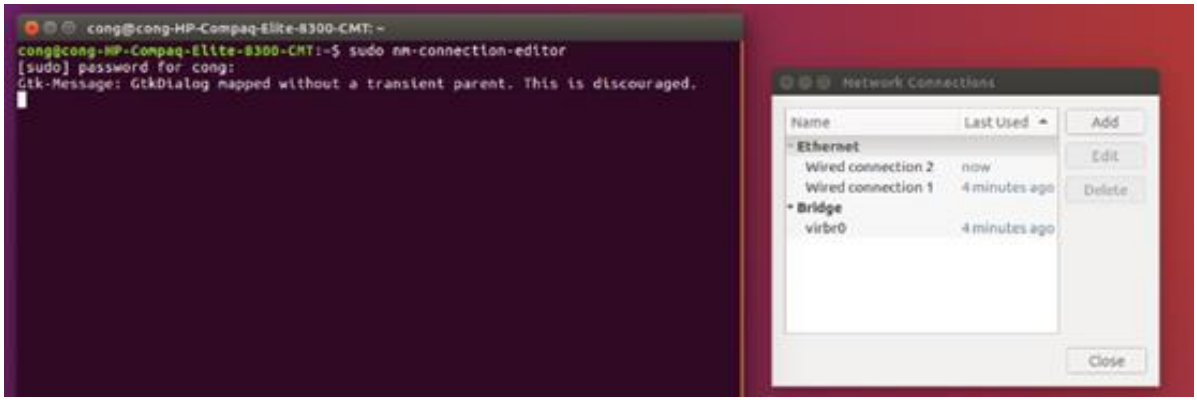

Figure 4. nm-commection-editor

Now, I will introduce how to create Linux bridge via nm-connection-editor step by step.

At the beginning, I clicked the “Add” button to create a bridge, next, I chose “Bridge” as a connection type and clicked the “Create” button. At this step, it is time to configure a bridge, encompassing a bridge name and bridge connections. By default, the bridge interface name is “bridge 0”. The aim of creating a bridge is to share the Ethernet interface via the bridge, so I added the Ethernet interface to the bridge. Here is the picture. (see Figure 5).

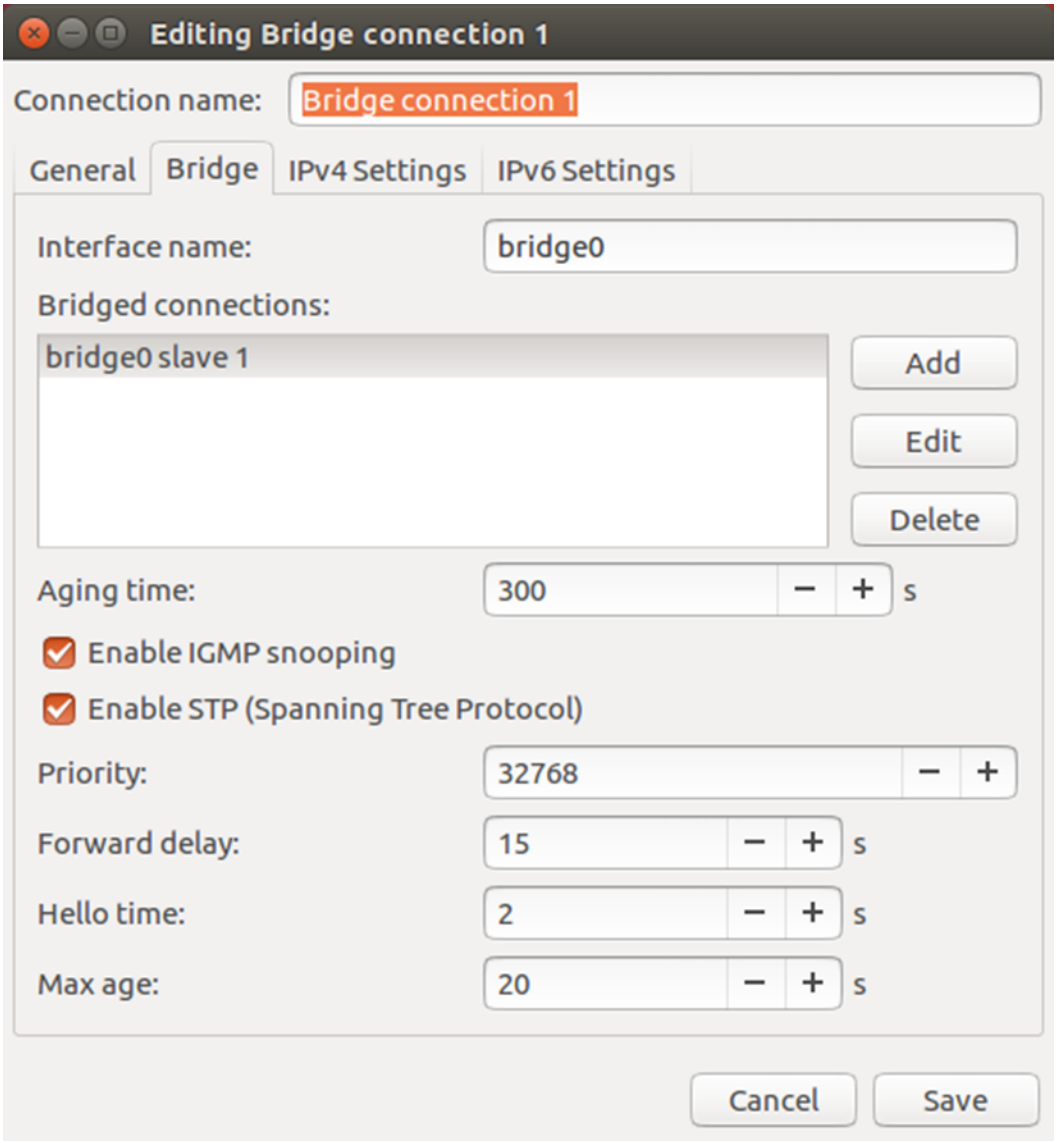

Figure 5. Bridged network

In “Device MAC address”, I should select the interface which I want to enslave into the bridge. I selected the eth0 interface MAC address. Then I moved to the IPv4 setting tab, I can configure either DHCP or static IP address for my bridge, I selected DHCP as my bridge. One important thing I need to point out is that if I configure with static IP address for the bridge, I should set the same IP address as the interface that I selected for my bridge. For example, I selected eth0 as my bridge interface, and I must assign the IP address of eth0 as my static IP address, and then I need just to save the configuration. Finally, I created the bridge for my Ubuntu host. Now I can see an additional “Bridge connection 1” created in the “Network Connection” window, shown in Figure 6.

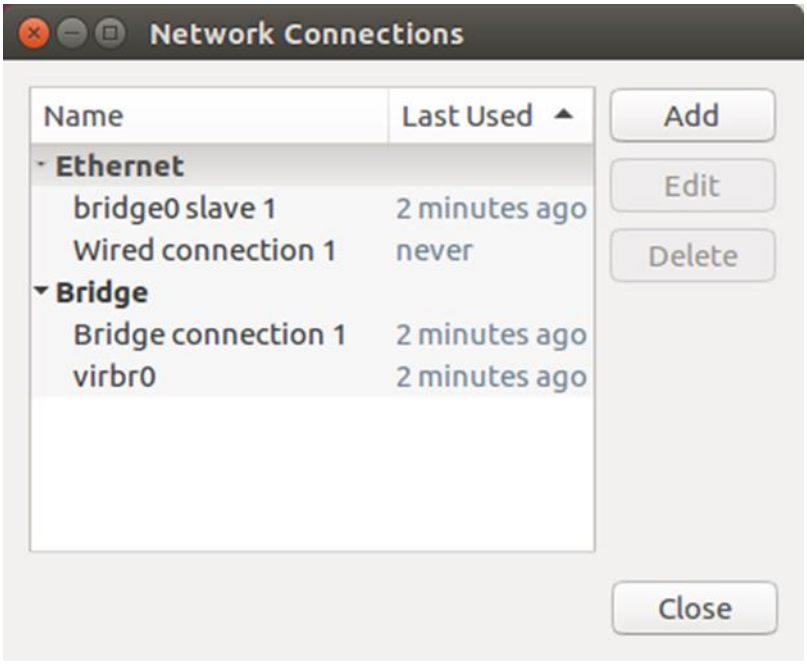

Figure 6. Bridge connection

Then I go ahead to delete the previous wired connection eth0. At this point, the bridge will be automatically activated. Since the IP address assigned to eth0 is taken over by the bridge. Once an IP address is assigned to the bridge, I will be connected back to my Ethernet interface via the bridge. Also I can check the list of available interfaces.

4.4 Managing KVM Remotely

Based on Ubuntu documentation (2016). In this part, the main idea is to manage or monitor the virtual machines through virtmanager tool: virt and libvirt virtualization library where I add the libvirtd users, and OpenSSH on both server and remote clients and connect OpenSSH, so that I can manage or monitor virtual machines remotely.

The virtmanager tool allows users to control or manage the virtual machines graphically, like creating, deleting, running, pausing, stopping the virtual machines. The virtmanager tool provides users with a graphical interface to communicate with KVM. Firstly, I installed the virtmanager tool with the command:

```
sudo apt-get install virt-manager
```

After installation, I can go to search for Virtual Machine Manager on the Ubuntu host and I will get it. Figure 7 shows the Virtual Machine Manager interface as below.

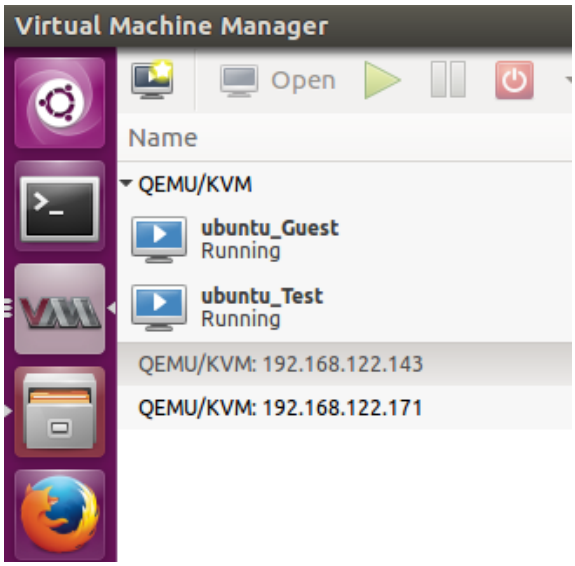

Figure 7. Virtual Machine Manager

With the Virtual Machine Manager, I can create virtual machines here. Here is the main simple process of how to create virtual machines on Virtual Machine Manager. At the beginning, to download a kind of ISO cd I prefer to run on my virtual machine. Secondly, click the item “Create a new virtual machine”, then I will get a wizard to guide me through the rest of my VM creation, I just need to enter my virtual machine details, like hostname, RAM, CPU and so on.

Next, it is time to install another virtmanager tool: libvirt, which is a virtual library to manage the available hypervisors. Then I add username into the group libvirtd on both Ubuntu host and virtual machines, using the command:

sudo adduser name libvirtd

I added a user called “guest” into my Ubuntu host and one of my virtual machines, Ubuntu_Guest, and the command is

sudo adduser guest libvirtd

However, there is one thing I should do before adding the “guest” user to the group libvirtd and it is that I must add the user “guest” to the group on Ubuntu host and that virtual machine- Ub-untu_Guest. As only the user “guest” belongs to the group Ubuntu host and virtual machine, I could successfully to add “guest” into the group libvirtd on my host and virtual machine, or it will refuse to be added into the group libvirtd in the right way. To add the user “guest” into group with the command: ***sudo adduser guest***

I will provide a picture as below to prove that I have successfully added the “guest” into the group libvirtd. (see Figure 8).

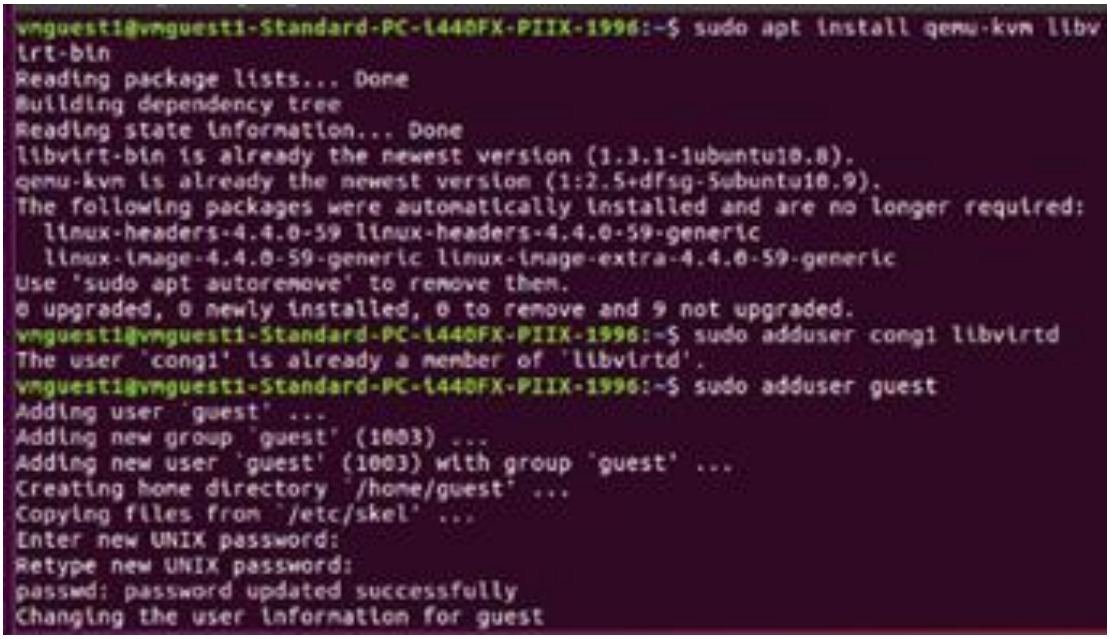

Figure 8. Adding a user into libvirtd

Thirdly, I started to install OpenSSH between Ubuntu host and Ubuntu_Guest with the command:
sudo apt install openssh-client and ***sudo apt install openssh-server***.

After this step, I need to enable the firewall UFW and check the firewall states, and then to allow the SSH connection, the commands I used well as follows:
sudo ufw enable sudo ufw allow 22

.

Everything done, I could manage or monitor the virtual machine, Ubuntu_Guest remotely. And I will explain how to launch KVM/VirtManager and connect with virtual machines in detail.

After getting access to KVM/VirManager, I need to click “File” -> “Add Connection”, and the interface shown in Figure 9 and 10 will appear.

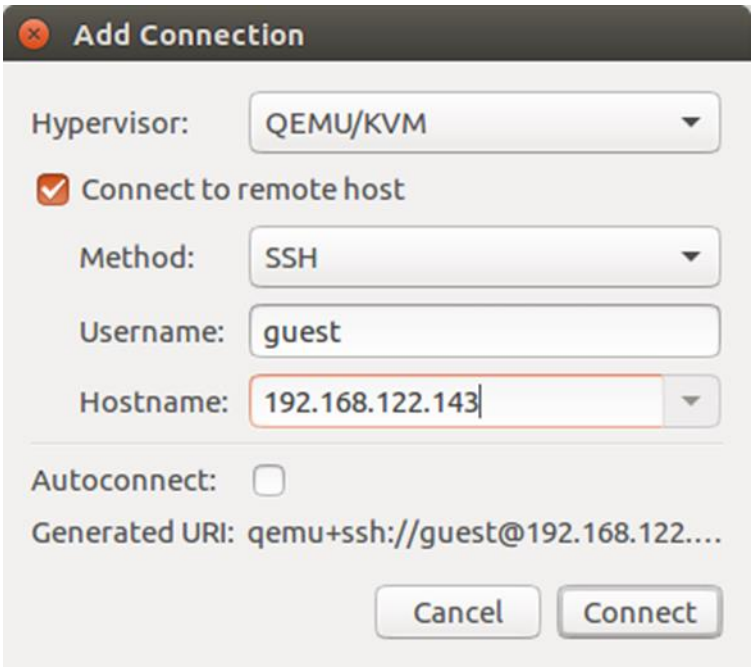

Figure 9. Add connection

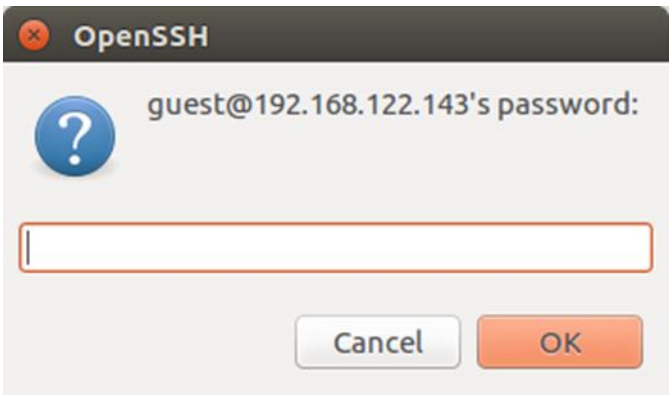

Figure 10. OpenSSH connection

Next, I select the item “Connect to remote host”, fill in Method: SSH, add “guest” and IP address 192.168.122.143 as Username and Hostname, input the password of OpenSSH to connect the virtual machine-“guest”, shown in Figure 11.

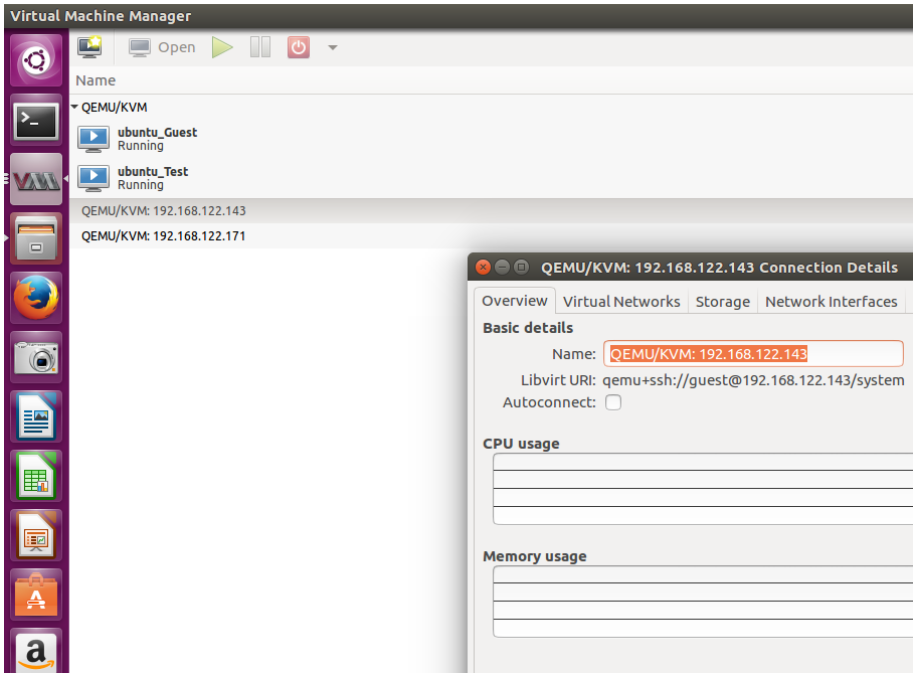

Figure 11. Connect to remote host

As the OpenSSH connection is setting up, I can remotely manage or monitor the virtual machines now. For example, I can check some basic things on the virtual machine-Ubuntu_Guest, like CPU usage, Memory usage, VM running states and so on, at the same time, I can also create storage volume, create a new virtual network, manage the states of VMs etc. With KVM/VirtManager, I can create any number of virtual machines, as long as there is enough storage space. With OpenSSH, I could manage or monitor my virtual machines remotely. It is a pretty effective way to handle my virtual machines. (see Figure 12).

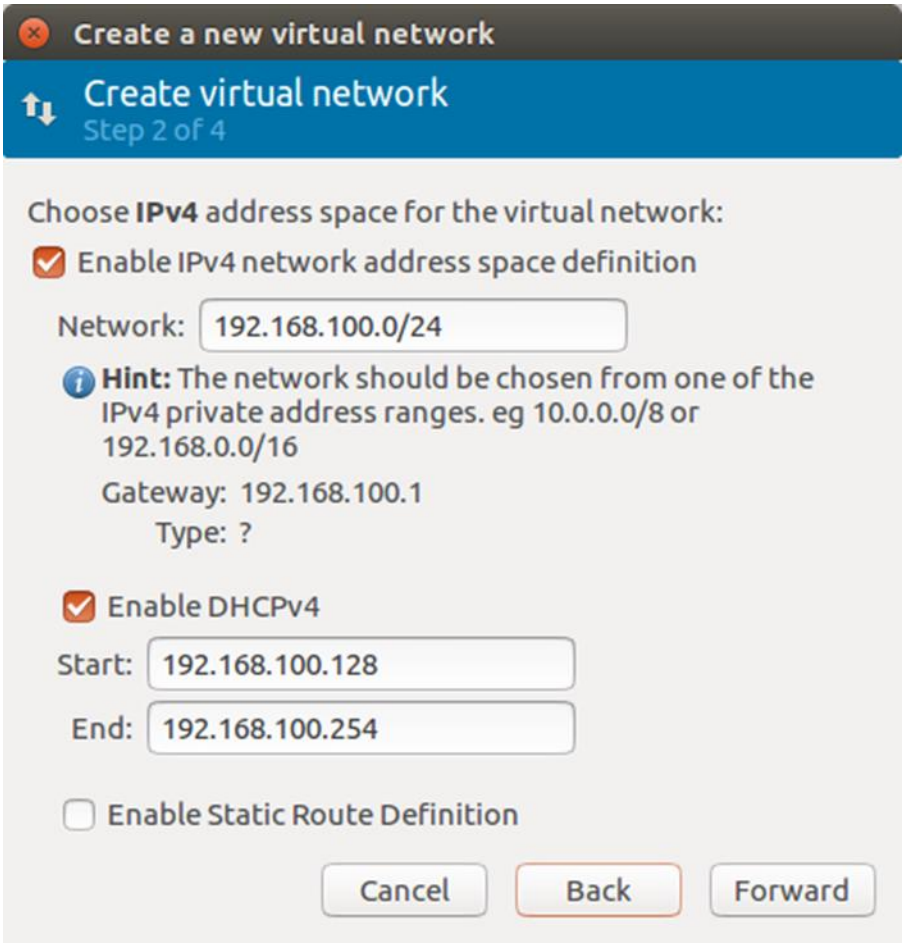

Figure 12. Creating a virtual network

4.5 Live KVM Migration With Virsh

Firstly, I created another virtual machine called “generic”. Then with the command:

virsh list

to check the virtual machine running state. (see Figure 13).

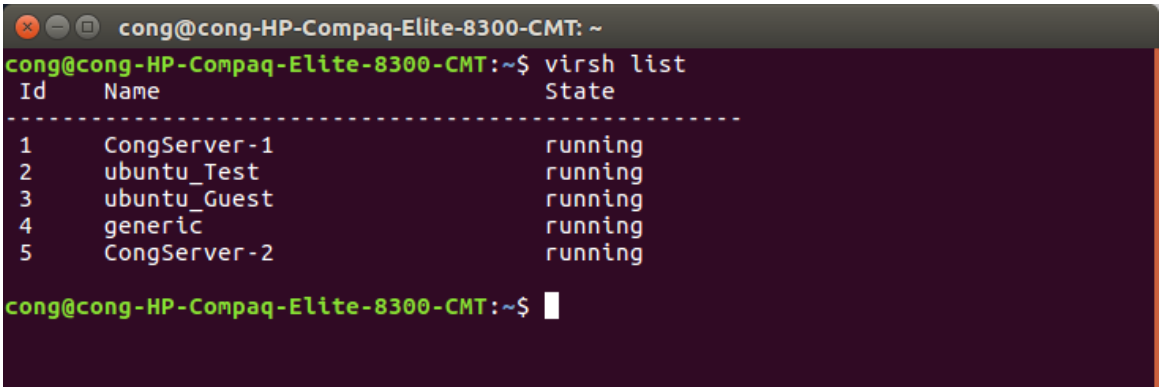

Figure 13. Virsh list

The virsh list shows that we can see that ”generic” is in the running state. Next, I installed and started network file system (NFS) with the command ***sudo apt install nfs-kernel-server*** and ***sudo systemctl start nfs-kernel-server.service***

on my physical host. Then, I started to edit the iptables with the following commands:

sudo iptables -P INPUT ACCEPT

sudo iptables -P OUTPUT ACCEPT

These commands ensure the default policy on INPUT and OUTPUT chains are set to ACCEPT.

I installed the Ubuntu operating system to another physical host (destination host) with the same version (16.04 LTS) as my host. The destination host has IP address 172.16.1.123. Then, I installed OpenSSH and the libvirt tool in the destination host and added the member “licong” to the libvirtd group in both the host and destination host. Next, I connected my host and the destination host with OpenSSH. Everything was done. I could migrate virtual machine-“generic” from my host to the destination host. There are two ways to carry out live migration, one way is to type the command:

 # virsh migrate --live generic qemu+ssh://li@li-HP-Compag-Elite-8300-CMT/system”.

The other way is to use graphical user interface (GUI) where fill in the corresponding information and migrate. It’s shown in Figure 14 and 15.

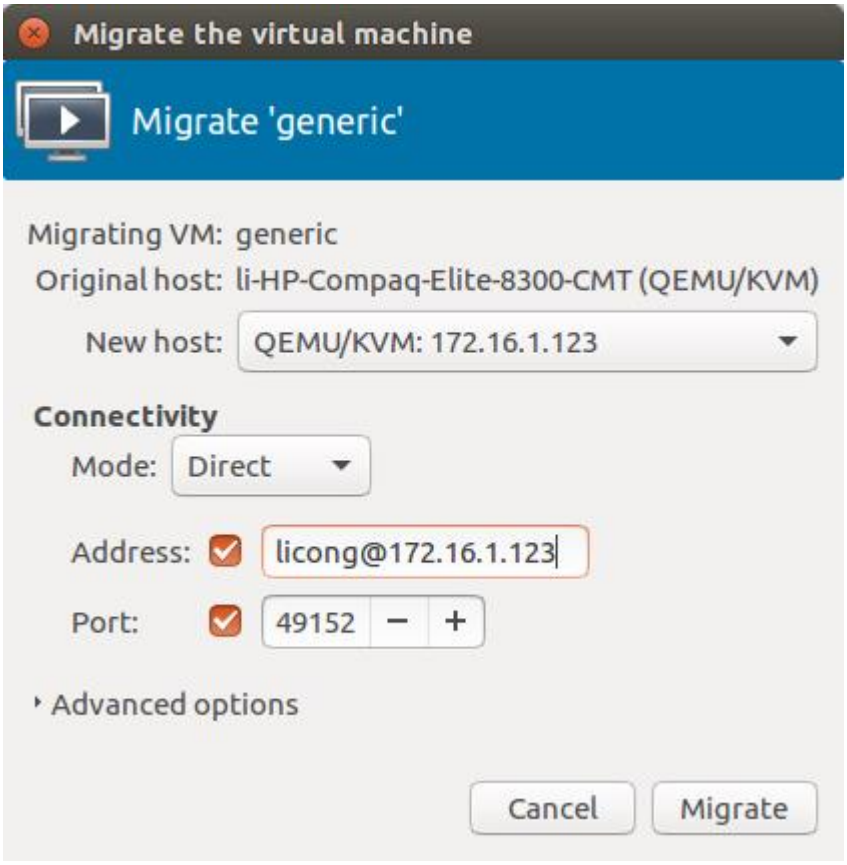

Figure 14. Migration connectivity

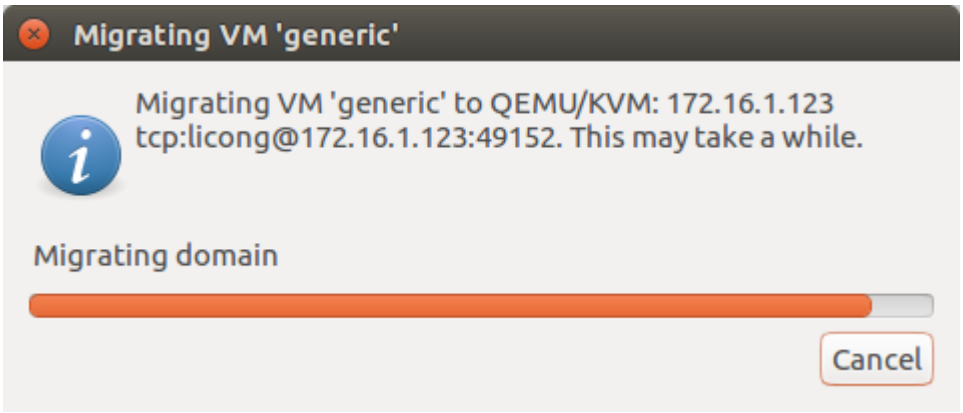

Figure 15. Live migration

After live migration, we can go to check the virsh list on host and destination. On the host, “generic” should disappear and on the destination host, the “generic” should be there. (see Figure 16 and 17).

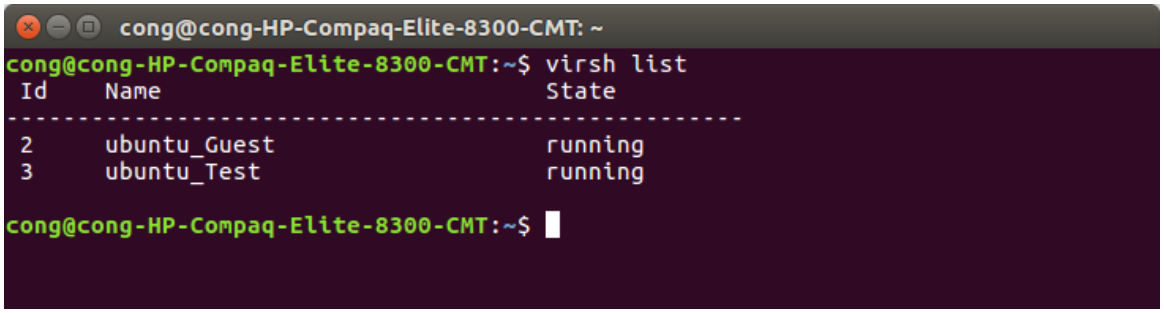

Figure 16. Virsh list on host

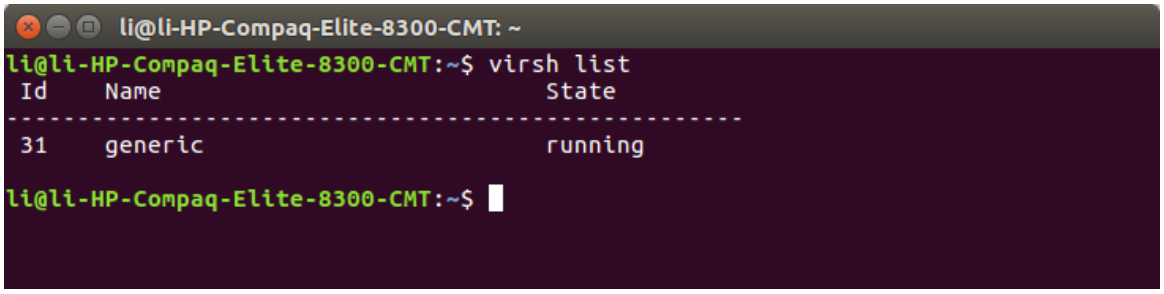

Figure 17. Virsh list on destination host

After that, I did the live migration from the destination host back to my host and it was successful as well. (see Figure 18).

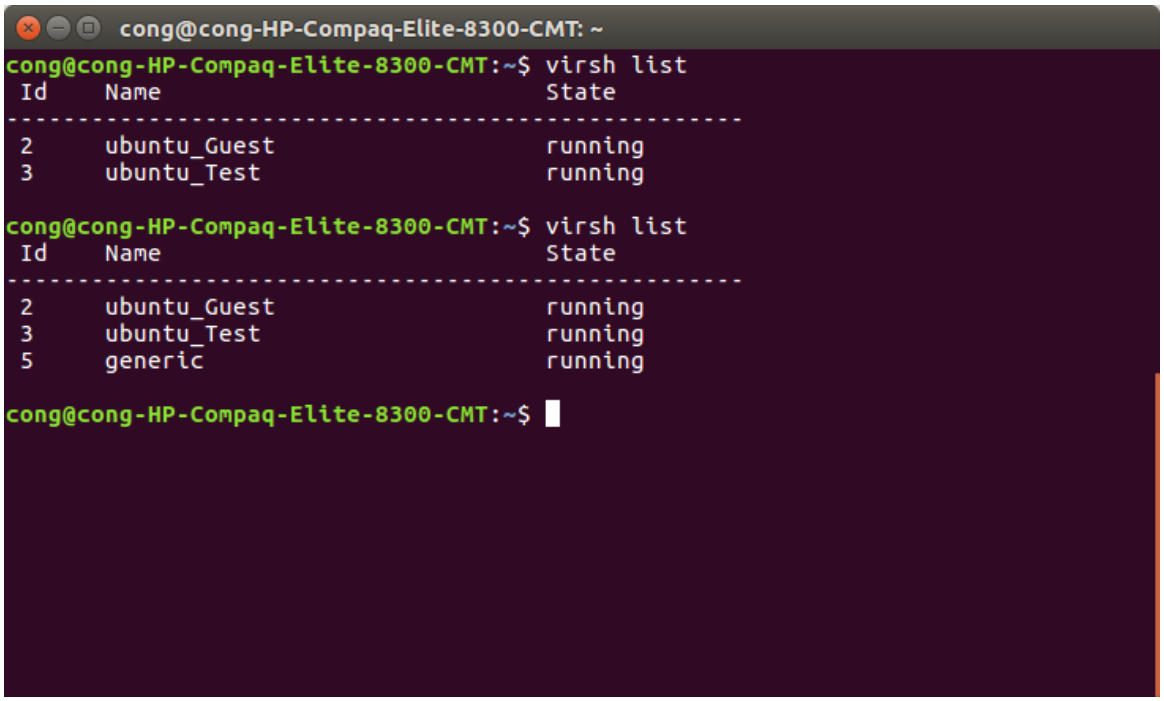

Figure 18. ”generic” is running on the host

4.6 DRBD Pacemaker HA Cluster

In this part, based on THE URBAN PENGUIN (2016). I will demonstrates about how to build up Distributed Replicated Block Device (DRBD) Pacemaker HA cluster in detail.

I installed two Ubuntu servers- Ubuntu 16.04.2 LTS on my virtual machine. The two servers' names are Cong1 and Cong2 seperately. Then I should create a partition of the same size on both servers that we can replicate data. Here I used disk- sda5 to replicate data. We must make sure that sda5 is clear with the commands:

sudo dd if=/dev/zero of=/dev/sda5

It takes some time to initialize the disk. (See Figure 19).

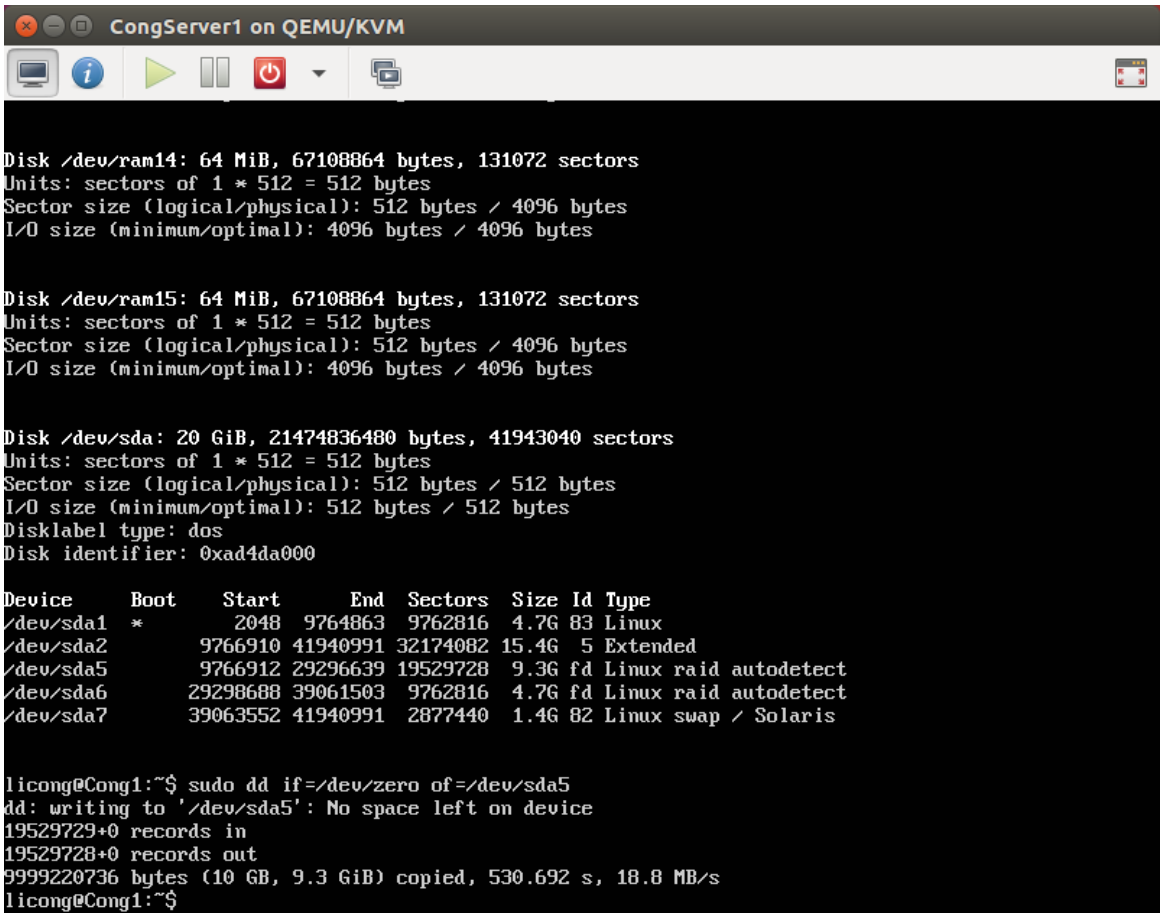

Figure 19. Initialize disk sda5

It was time to install drbd application with commands*:*

sudo apt-get install -y drbd8-utils

After installation, we should ensure that we can resolve my servernames to the IP Address that we would use in the cluster. I could open the file with commands:

sudo nano /etc/hosts

to edit the IP Address. In my cluster, I set the IP address as following on both servers Cong1 and Cong2:

127.0.0.1 local host

192.168.122. 235 Cong1

192.168.122.54 Cong2

Next, I need to edit /etc/drbd.conf on Cong1 and Cong2. Here is the configuration as bellow: (See Figure 20).

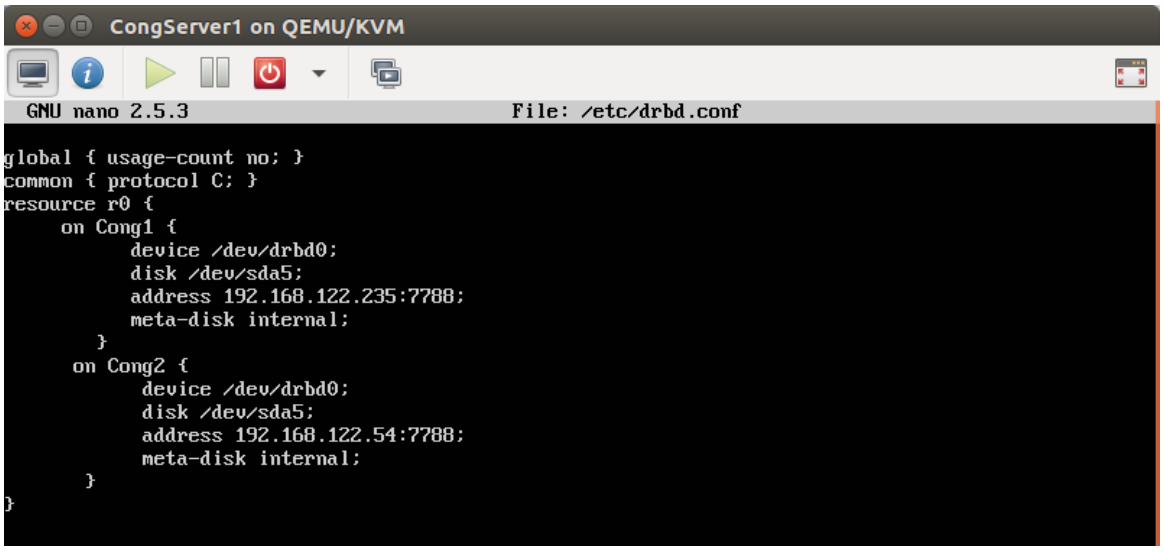

Figure 20. File:/etc/drbd.conf configuration

Then, we need to load the kernel module on Cong1 and Cong2 with:

sudo modprobe drbd

to create the mirror device and bring the mirror device online with the commands:

sudo drbdadm create-md r0 and ***sudo drbdadm up r0***

I would list the new block device with lsblk and we can see the drbd0 has created. (See Figure 21).

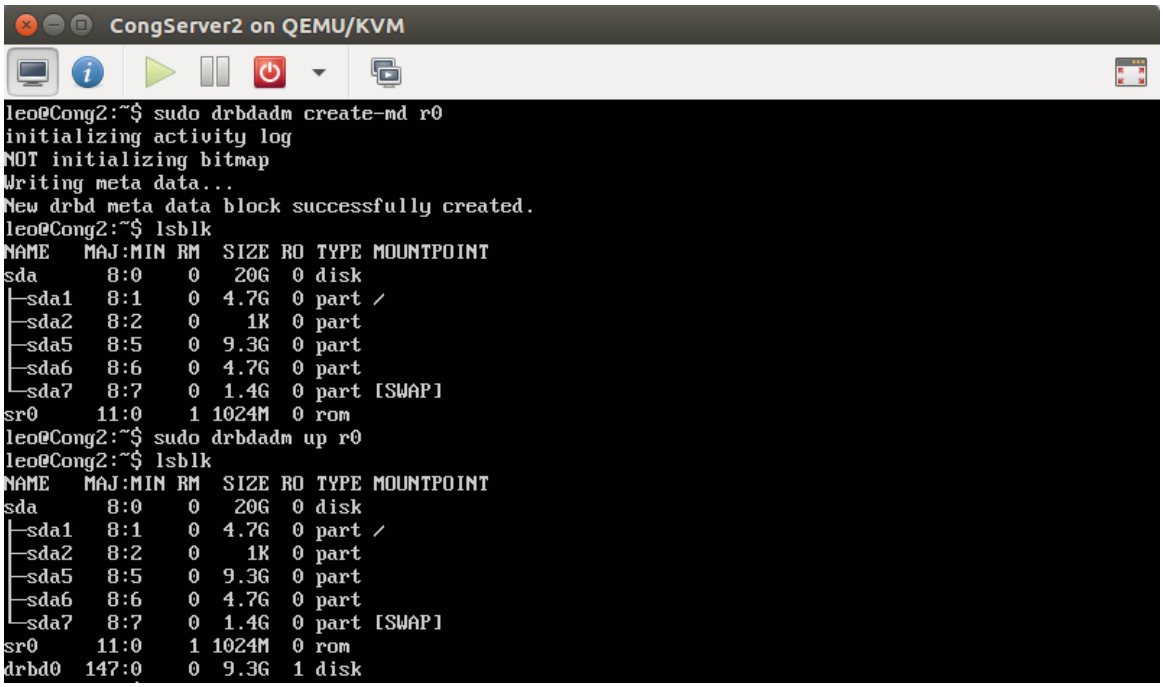

Figure 21. lsblk list table

Next step is to assign a Primary role on either Cong1 or Cong2. I select Cong1 node as my Primary role with the command:

sudo drbdadm -- --overwrite-data-of-peer primary r0/0.

This processing would take some time to finish. We can view the progress:

sudo watch cat /proc/drbd.

From the following picture, we can see the primary node- Cong1 set up successfully, based on the command: Primary/Secondary UpToDate/UpToDate (See Figure 22).

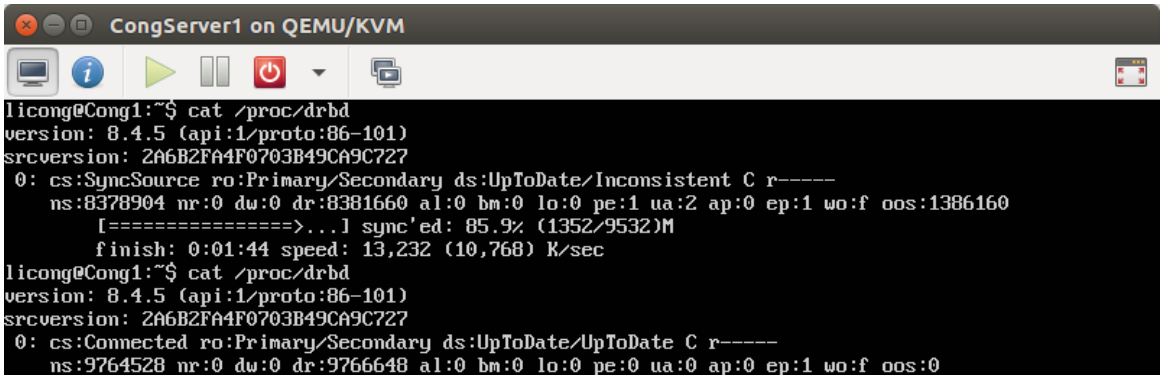

Figure 22. Set up Primary role

After we complete from the Primary node we can create a file system on mirror device-drbd0 of Primary node and mount it with the following commands:

sudo mkfs.ext4 /dev/drbd0

sudo mkdir -p /var/www/html

sudo mount /dev/drbd0 /var/www/html

Then I can list the content of /var/www/html, I found nothing but the file system " lost and file" then I would add some data in /var/www/html with the commands:

sudo touch /var/www/html/congfile1

Then I would umount /var/www/html, switch Cong1 server from Primary node to Secondary node. Next, I went to Cong2 server and set it as Primary node, where I created /var/www/html and mount it to the mirror device drbd0 and I can list /var/www/html, now it should be found the file content- congfile1 here. Here are the related pictures as following. (See Figure 23 & 24).

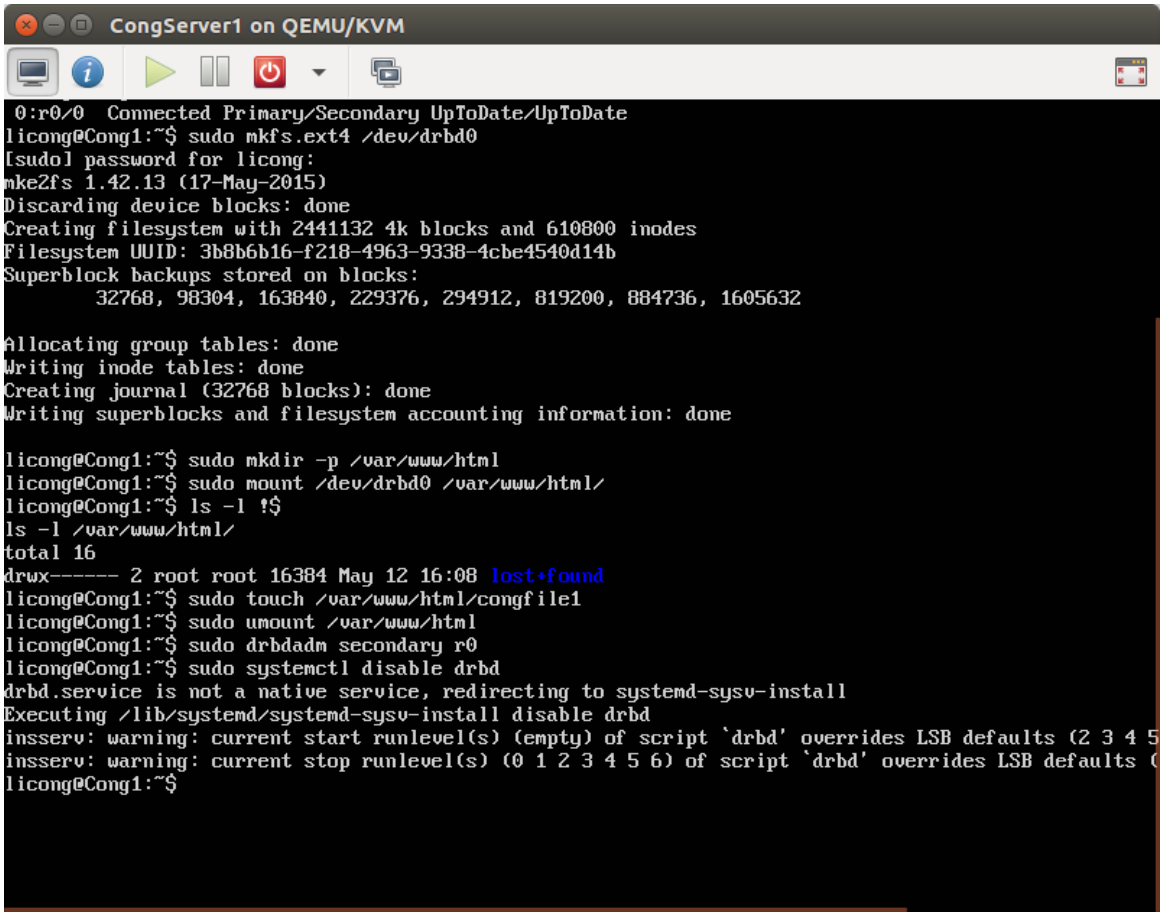

Figure 23. Umount /var/www/html

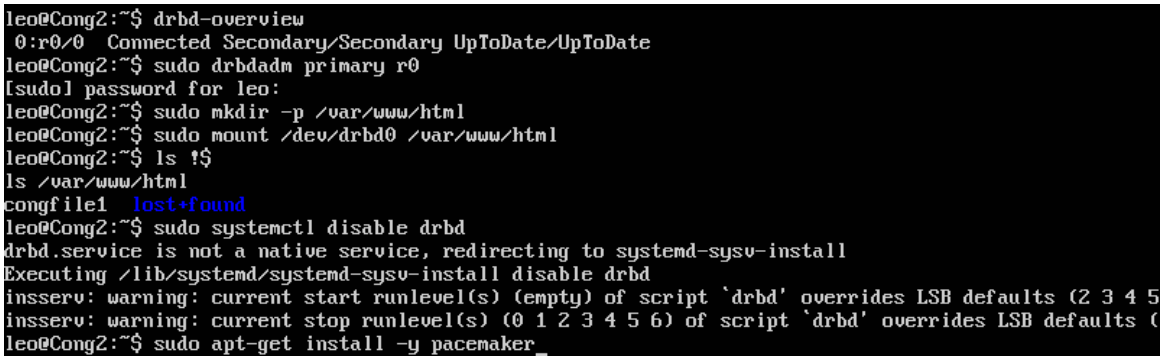

Figure 24. Set drbd primary r0 on Cong2

From the two pictures above, we could find the file-congfile1 on Cong2 serve that proves I created HA cluster successfully. However this is not an effective method to provide us HA cluster. We should do everything manually, like if we create a file or whatever and we want to have the other node service this we have to un-mount the device, change primary to secondary, change secondary to primary and mount the device. That ‘s the reason why Cluster Resource Manager-Pacemaker comes into play to manage this for us.

Before installing Pacemaker. I must disable DRBD on both servers with commands: sudo systemctl disable drbd. Then, I installed the Pacemaker on both servers- Cong1 and Cong2 with the command: sudo apt-get install -y pacemaker. Next, I edited Corosync that is used to synchronize the Pacemaker cluster details. Here is the related configuration in /etc/corosync/corosync.conf file on both devices. (See Figure 25).

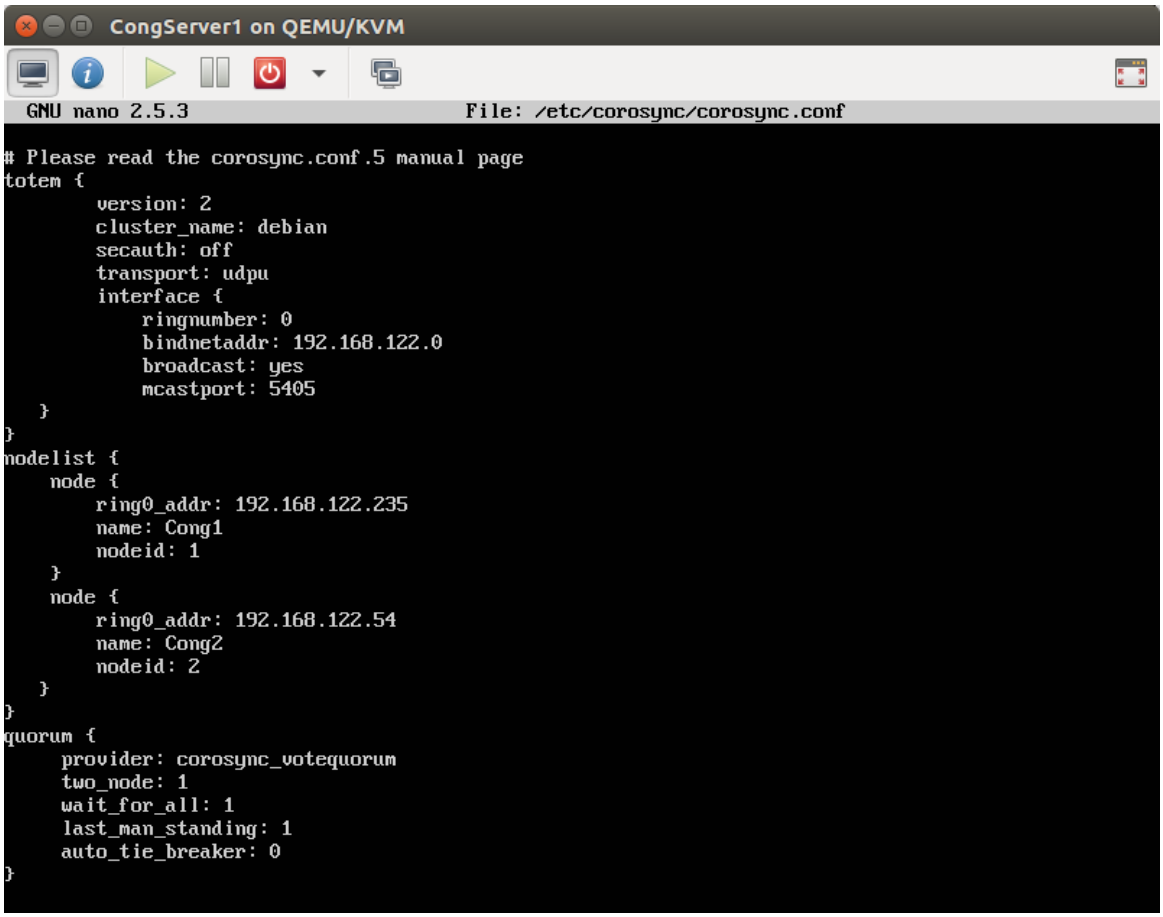

Figure 25. /etc/corosync/corosync.conf configuration

We can restart corosync and start pacemaker on both nodes:

sudo systemctl restart corosync

sudo systemctl start pacemaker

On Cong1 or Cong2, I can configure some properties for a simple nodes first. Typing the command crm configure will take it to an interactive prompt and I need do this configuration on either Cong1 or Cong2. Here I set the properties on

Cong1. (See Figure 26). I then create the drbd resource which is further controlled by the Master Slave set. This makes sure that only one Master role is assigned to the drbd resource. I then create the file system resource to mount the drbd disk and make sure that they run together with the colocation command. The order is maintained with the order command.

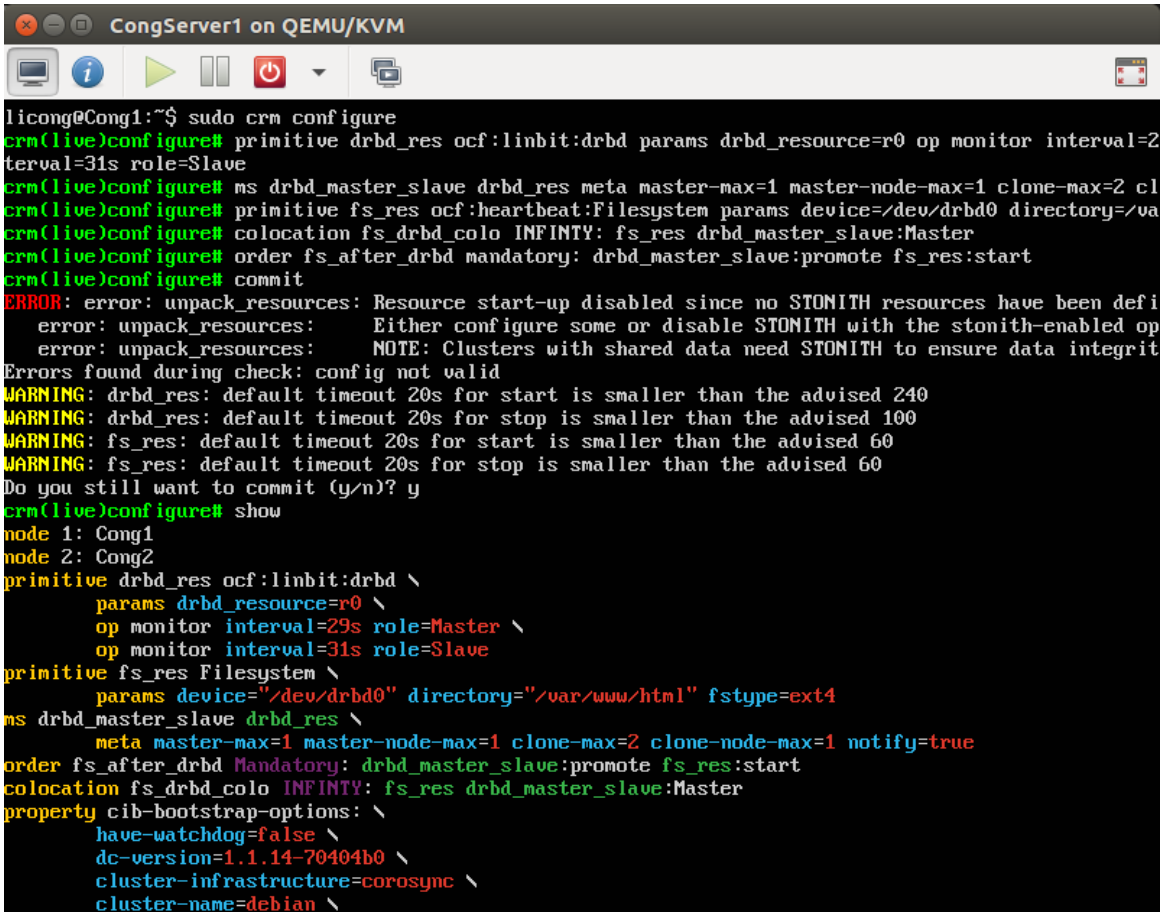

Figure 26. crm(live) configure

From Figure 26, we can see there is a error report: unpack_resources: Resource start-up disabled since no STONITH resources have been defined and etc. According to Margaret (2013), the STONITH (Shoot The Other Node In The Head) is a Linux service for maintaining the integrity of nodes in a high-availability (HA) cluster.To solve this problem, according to Debian Heartbeat (2017), I need to disable STONITH, we set the stonith-enabled cluster option to false:

crm configure property stonith-enabled=false

crm_verify -L -V.

After finishing the configuration. We should now get a HA cluster that will manage the mounting automatically. And on either Cong1 or Cong2, I can use the command:

sudo crm status

to view the cluster come online. (See Figure 27).

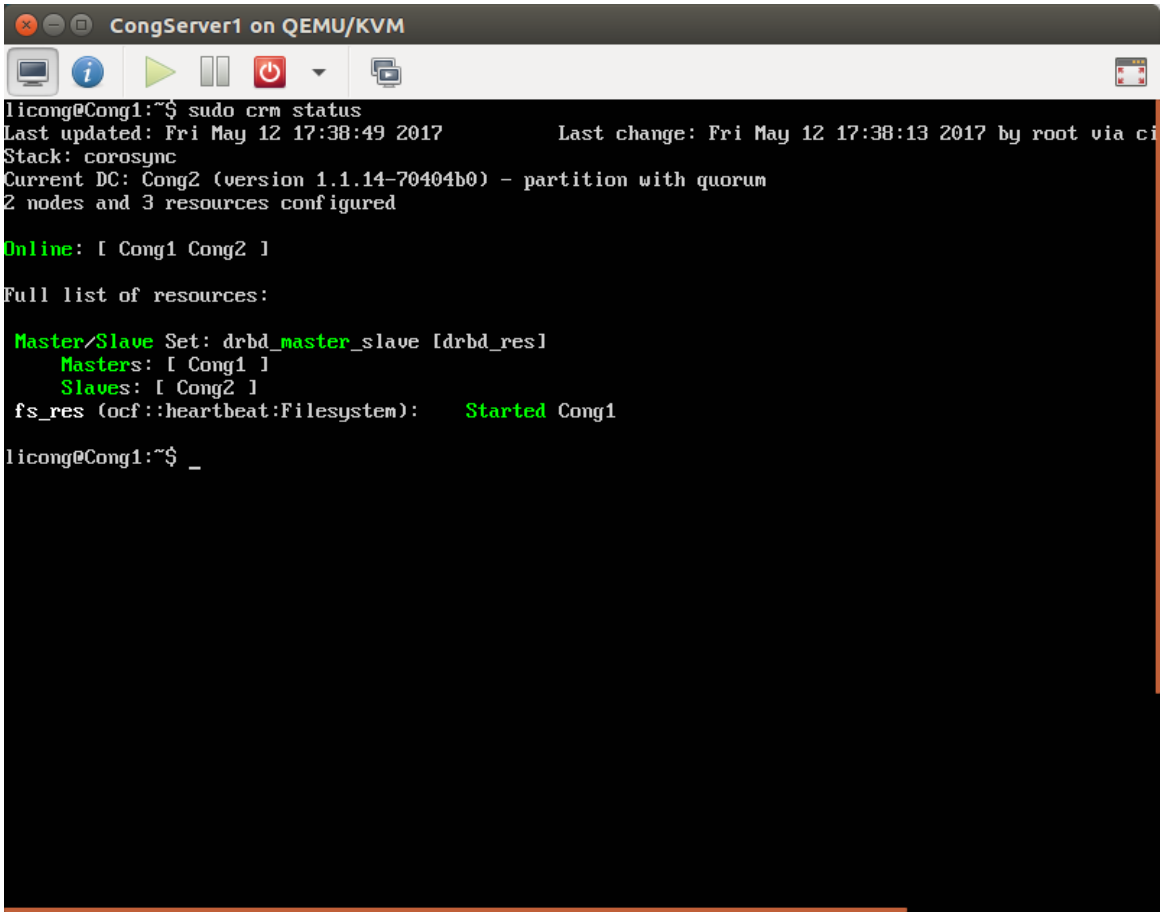

Figure 27. Clusters come online

To check if the two nodes work fine, let's do some test. From the Figure 27, we can view that Masters: Cong1 and Slaves: Cong2 and both nodes are online. Now started situation is Cong1. On Server- Cong1, with the command:

ls /var/www/html

we can find the file- congfile1 here. I will reboot the Cong1 then see if Cong2 will act as the role of Cong1 and if we can find the file-confile1 on Cong2 as well. If we can find the file- congfile1 on Cong2 that proves the DRBD Pacemaker HA Cluster works successfully.(See Figure 28 & 29).

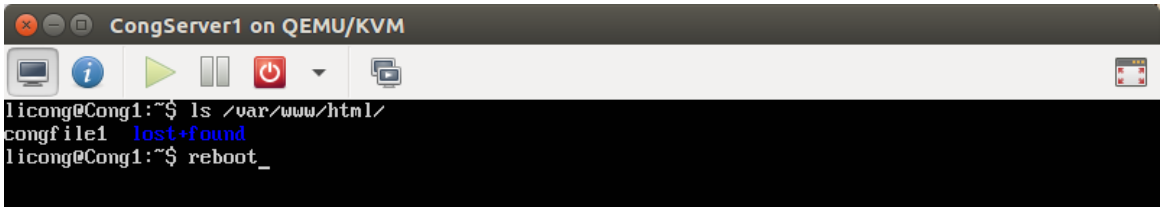

Figure 28. Reboot Cong1

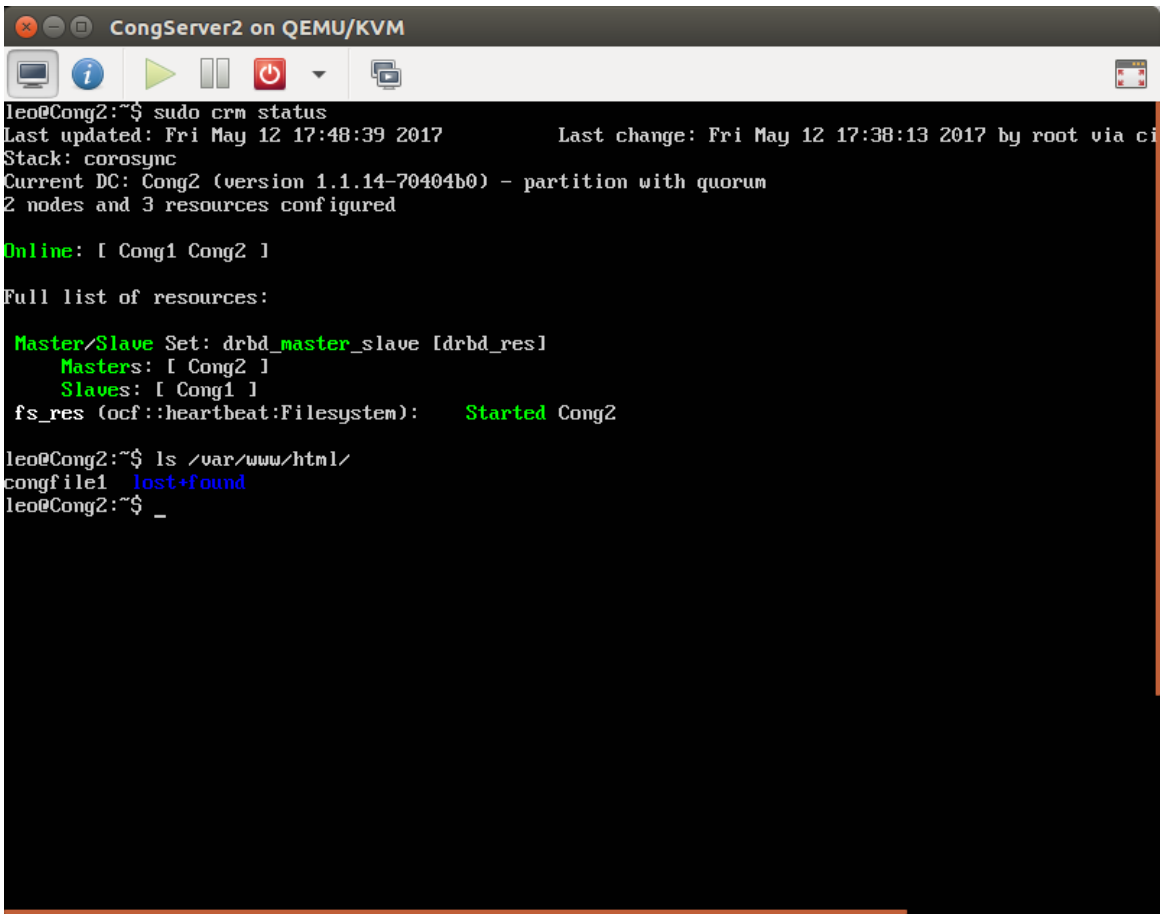

Figure 29. congfile1 found on Cong2

From Figure 28 and 29. We can see that the server- Cong2 now is Master and Cong1 now Slaves. The started situation is now Cong2. We can also find the file-congfile1 on Cong2. Everything works perfectly, which means I created DRBD Pacemaker HA Cluster successfully.

5 CONCLUSION

Virtual machines allow users to run multiple hosts on one physical host that can save a lot of money for companies. At the same time, consolidating servers onto fewer machines means that less physical space is required for computer systems. An increasing numbers of companies prefer to install multiple virtual machines on one physical host rather than to purchase multiple physical hosts. These are some reasons why the virtual machine companies are becoming more and more popular in the world. Virtualization has always been a complex technology. However, the virtualization companies are developing and updating the new virtualization technology consistently. For instance, based on Microsoft (2017), Microsoft released its various virtualization products, such as Hyper-V, App-V, and MED-V. According to VMware (2017), VMware provides people with the new technology, like Software-Defined Data Center, vCloud Air Network, vSphere, vCenter Server etc. Another example is Red Hat company. Based on Dan (2017), Red Hat company increasing the investing value in the KVM project and integrating VM technology more tightly into other products to make it easier for cooperations to adopt the complete Red Hat software environment.

From these three examples, each virtualization company has been developing and investing in virtualization technology. It is clear that virtualization technology plays a significant role in the enterprises.

My project was KVM and everything worked based on the Linux operating system and virtualization environment, so that I introduced same basic things at the beginning like virtualization environment, how to install Linux operating system and how to install the KVM virtual machine and some related remote management tools and applictions, like libvirtd, virsh, OpenSSH etc.

Next, I set the bridged network on my virtual machines. Bridged network is a technology that shares a real Ethernet device with virtual machines (VMs) and each virtual machine acts as a physical host. Bridged network is a significant technology in the KVM virtualization environment.

With the bridged network and series of remote management tools, I could connect virtual machines and manage them remotely, such as view virtual machines' basic details like cpu and memory usage etc, create a virtual network and etc. Remote management provides people with a convenient and effective method to manage virtual machines.

KVM live migration allows virtual machines to move from one physical host to another without shutting down the virtual machines. Two basic migration methods are offline migration and live migration. The greatest difference is that with offline migration, the virtual machine which is used for migration should be powered off. However, with online migration, the virtual machine used for migration allows to be powered on. KVM migration provides people with a easy way to move the virtual machines from one host to another rather than install virtual machines every time to a new host. In case the host system is crashed, the virtual machines can still be moved to another host and keep working. Live migration is another important technology in virtualization environment.

The last part is HA cluster. HA cluster provides a higher level of availability, reliability, and scalability for clients. I tried two methods to create the HA cluster. One way is that I created a manual HA cluster. It works fine, however, you should manual control the servers Primary node and Secondary node, mount and umount between two different nodes. It is not an effective HA cluster. For the second method, I tried DRBD Pacemaker HA cluster which is a automatic control HA cluster which could provide us with a quite and an effective HA cluster. HA cluster was the most challenging part in my project. When I was doing HA cluster, I met a lot of challenges and it took me a lot of time to do some research. However, I made cluster work perfectly finally.

To sum up, I met all kinds of chanllenges, when I was doing this project, however, I solved all the problem and figured everything out. Of course there are many better ideas to be improved in this project. However, I learnt a lot of new and useful things through doing this project.

REFERENCES

Application Virtualization. 2017. Microsoft. WWW document. Available at: https://technet.microsoft.com/en-us/library/hh826068.aspx [Accessed May 15 2017].

Advantages and Disadvantages of Server Clustering. 2016. Industry news.WWW document. Updated 25 May 2011. Available at: https://www.eukhost.com/blog/webhosting/advantages-and-disadvantages-of-server-clustering/ [Accessed 13 April 2017].

Andrew A. 1996. The Network File System. WWW document. Available at: http://www.tldp.org/LDP/nag/node140.html [Accessed 2 April 2017].

Canonical Ltd. Ubuntu. 2017. Download Ubuntu Desktop. WWW document. Available at: https://www.ubuntu.com/download/desktop [Accessed 15 January 2017].

Cluster Basics. 2017. Red Hat, Inc. WWW document. Available at: https://www.centos.org/docs/5/html/Cluster_Suite_Overview/s1-clstr-basics-CSO.html [Accessed 5 May 2017].

Download Ubuntu Desktop. 2017. Canonical Ltd. WWW document. Available at: https://www.ubuntu.com/download/desktop [Accessed 1 January 2017].

Dan N. 2015. How to configure a Linux bridge with Network Manager on Ubuntu. WWW document. Available at: http://ask.xmodulo.com/configure-linux-bridge-network-manager-ubuntu.html [Accessed time January 20 2017].

DRBD Pacemaker HA Cluster on Ubuntu 16.04. 2016. THE URBAN PENGUIN. WWW document. Available at: https://www.theurbanpenguin.com/drbd-pacemaker-ha-cluster-ubuntu-16-04/ [Accessed 5 May 2017].

DRBD. 2017. Ubuntu documentation. WWW document. Available at: https://help.ubuntu.com/lts/serverguide/drbd.html [Accessed 8 May 2017].

Debian Heartbeat: deal with STONITH errors when configuring CRM cluster. 2014. 2017. WWW document. Available at: http://docs.gz.ro/CRM-cluster-STONITH.html [Accessed 12 May 2017].

Daniel P. 2013. Manage virtual machines with virt-manager. WWW document. Available at: https://virt-manager.org/ [Accessed 15 February 2017].

Dan K. 2016. Red Hat Virtualization 4: An Overview. WWW document. Available at: https://virtualizationreview.com/articles/2016/09/12/red-hat-virtualization-4-overview.aspx [Accessed May 15 2017].

FORWARD and NAT Rules. 2017. Red Hat, Inc. WWW document. Available at: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/4/html/Security_Guide/s1-firewall-ipt-fwd.html [Accessed 15 March 2017].

Hyper-V overview. 2017. WWW document. Available at: https://technet.microsoft.com/en-us/library/hh831531(v=ws.11).aspx [Accessed January 15 2017].

Hyper-V Installation Prerequisites. 2017. Microsoft. WWW document. Available at: https://technet.microsoft.com/en-us/library/cc731898(v=ws.11).aspx [Accessed 15 January 2017].

Hardware Considerations for Hyper-V in Windows Server 2008. 2017. Microsoft. WWW document. Available at: https://technet.microsoft.com/en-us/library/cc816844(v=ws.10).aspx [Accessed 20 January 2017].

Hoffman C. 2012. How to Install KVM and Create Virtual Machines on Ubuntu. WWW document. Available at: https://www.howtogeek.com/117635/how-to-install-kvm-and-create-virtual-machines-on-ubuntu/ [Accessed 8 February 2017].

How To Make A Live Migration Of A KVM Virtual Machine Using Virsh And SSH. 2010. WWW document. Available at: https://acidborg.wordpress.com/2010/02/22/how-to-make-a-live-migration-of-a-kvm-virtual-machine-using-virsh-and-ssh/ [Accessed time 5 March 2017].

IptablesHowTo. 2017. Ubuntu documentation. WWW document. Available at: https://help.ubuntu.com/community/IptablesHowTo [Accessed 15 March 2017].

Jamie N. 2015. Bridged network. WWW document. Available at: https://jamielinux.com/docs/libvirt-networking-handbook/bridged-network.html [Accessed 15 February 2017].

Juutilainen M. 2017. Why Virtualize. WWW document. Available at: https://moodle.xamk.fi/pluginfile.php/891804/mod_resource/content/0/02%20Virtualization%20benefits%20and%20risks.pdf [Accessed 5 February 2017].

Kernel Virtual Machine. 2016. Red Hat, Inc. WWW document. Available at: https://www.linux-kvm.org/page/Main_Page [Accessed time 2 January 2017].

KVM/Installation. 2016. Ubuntu documentation. WWW document. Available at: https://help.ubuntu.com/community/KVM/Installation#Pre-installation_checklist [Accessed time 2 January 2017].

Kernel-based Virtual Machine. 2017. Wikimedia Foundation, Inc. WWW document. Available at: https://en.wikipedia.org/wiki/Kernel-based_Virtual_Machine [Accessed 10 January 2017].

KVM/Networking. 2017. Ubuntu documentation. WWW document. Available at: https://help.ubuntu.com/community/KVM/Networking [Accessed time January 14 2017].

KVM/VirtManager. 2016. Ubuntu documentation. WWW document. Available at: https://help.ubuntu.com/community/KVM/VirtManager [Accessed 12 February 2017].

Kernel Virtual Machine. 2017. Red Hat, Inc. WWW document. Available at: https://www.linux-kvm.org/page/Migration [Accessed time 5 April 2017].

Korbin B. 2014. The Beginner's Guide to iptables, the Linux Firewall. WWW document. Available at: https://www.howtogeek.com/177621/the-beginners-guide-to-iptables-the-linux-firewall/ [Accessed time 10 March 2017].

KVM/Virsh. 2014. Ubuntu documentation. WWW document. Available at: https://help.ubuntu.com/community/KVM/Virsh [Accessed time 20 March 2017].

Kenneth H. 2010. Top 10 Virtualization Technology Companies. WWW document. Available at: http://www.serverwatch.com/article.php/3877576/Top-10-Virtualization-Technology-Companies.htm [Accessed 8 January 2017].

KVM/Networking. 2016. Ubuntu documentation. WWW document. Available at: https://help.ubuntu.com/community/KVM/Networking [Accessed 11 February 2017].

KVM or VMware: Why KVM is Right for the Times (Part 1 of 2). 2017. Scale Computing. WWW document. Available at: http://blog.scalecomputing.com/kvm-or-vmware-why-kvm-is-right-for-the-times-part-1-of-2/ [Accessed 8 February].

KVM live migration. 2017. Fedora documentation. WWW document. Available at: https://docs.fedoraproject.org/en-US/Fedora/13/html/Virtualization_Guide/chap-Virtualization-KVM_live_migration.html#item-Virtualization-Live_migration_requirements-Migration_requirements [Accessed 10 March 2017].

Live KVM Migration with virsh. 2017. Red Hat, Inc. WWW document. Available at: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Virtualization_Administration_Guide/sect-Virtualization-KVM_live_migration-Live_KVM_migration_with_virsh.html [Accessed time 10 April 2017].

Linux. 2017. Wikimedia Foundation, Inc. WWW document. Available at: https://en.wikipedia.org/wiki/Linux [Accessed 10 January 2017].

Margaret R. 2009. Intel VT (Virtualization Technology). WWW document. Available at: http://searchservervirtualization.techtarget.com/definition/Intel-VT [Accessed 21 December 2016].

Margaret R. 2009. AMD-V (AMD virtualization). WWW document. Available at: http://searchservervirtualization.techtarget.com/definition/AMD-V [Accessed 21 January 2017].

Margaret R. 2017. KVM hypervisor. WWW document. Available at: http://whatis.techtarget.com/definition/KVM-hypervisor [Accessed time January 12 2017].

M. Jones. 2012. Managing VMs with the Virtual Machine Manager. WWW document. Available at: https://www.ibm.com/developerworks/cloud/library/cl-managingvms/ [Accessed 13 February 2017].

Margaret R. 2016. Network File System (NFS). WWW document. Available at: http://searchenterprisedesktop.techtarget.com/definition/Network-File-System [Accessed time 12 March 2017].
Margaret R. 2013. High availability cluster (HA cluster). WWW document. Available at: http://searchdisasterrecovery.techtarget.com/definition/high-availability-cluster-HA-cluster [Accessed 2 May 2017].

Margaret R. 2016. Virtualization. WWW document. Available at: http://searchservervirtualization.techtarget.com/definition/virtualization [Accessed 16 January 2017].

Margaret R. 2017. KVM hypervisor. WWW document. Available at: http://whatis.techtarget.com/definition/KVM-hypervisor [Accessed 9 February 2017].

Margaret R. 2013. STONITH. WWW document. Available at: http://searchservervirtualization.techtarget.com/definition/STONITH-Shoot-The-Other-Node-In-The-Head [Accessed 18 May 2017].

Margaret R. 2013. High availability cluster. WWW document. Available at: http://searchdisasterrecovery.techtarget.com/definition/high-availability-cluster-HA-cluster [Accessed April 5 2017].

OpenSSH. 2017. Wikimedia Foundation, Inc. WWW document. Available at: https://en.wikipedia.org/wiki/OpenSSH [Accessed time 21 February 2017].

OpenSSH. 2017. OpenSSH. WWW document. Available at: https://www.openssh.com/ [Accessed 15 Februar 2017].

Operating System - Quick Guide. 2017. Tutorialspoint. WWW document. Available at: https://www.tutorialspoint.com/operating_system/os_quick_guide.htm [Accessed January 5 2017].

Products. VMware. 2017. WWW document. Available at: http://www.vmware.com/ [Accessed May 15 2017].

Prabesh S. 2013. What are all the advantages of using Linux rather than Microsoft Windows. WWW document. Available at: https://www.quora.com/What-are-all-the-advantages-of-using-Linux-rather-than-Microsoft-Windows [Accessed time 12 January 2017].

VMware. 2017. Wikimedia Foundation, Inc. WWW document. Available at: https://en.wikipedia.org/wiki/VMware [Accessed 12 January 2017].

Virtual machine migration. 2017. Techopedia Inc. WWW document. https://www.techopedia.com/definition/15033/virtual-machine-migration-vm-migration [Accessed 5 March 2017].

What is a server cluster. 2003. Microsoft. WWW document. Available at: https://technet.microsoft.com/en-us/library/cc785197(v=ws.10).aspx [Accessed 17 March 2017].

,