Graduation Report

Students	Nhat Tran – 427363	Date **06 – 04 – 2020**
Institution / Academy	SUAS - Academy of Creative Technology	
Course	HBO-IT – Graduation project	
Graduation teacher	Tonny Lievers	
Company supervisor	Anh Le	

TINOHOST

Start Your Business

TABLE OF CONTENTS

This page is intentionally left blank.

LIST OF ABBREVIATION

AWS	Amazon Web Service
CC	Cloud computing
GCP	Google Cloud Platform
GUI	Graphical User Interface
HA	High Availability
I/O	Input/ Output
IaaS	Infrastructure as a Service
PaaS	Platform as a Service
SaaS	Software as a Service
SLA	Service Level Agreement
NIST	National Institute of Standard and Technology
KVM	Kernel-based Virtual Machine
OS	Operation System
VM	Virtual Machine
VMM	Virtual Machine Monitor
VPS	Virtual Private Serve

PREFACE AND ACKNOWLEDGEMENT

For six consecutive months, from September 2019 till March 2020, did I a graduation assignment at a corporation which is named TinoHost in Viet Nam. As a Vietnamese, I returned to home country to gain experience in a demographically local to me ICT company. TinoHost is a web hosting company which is in the top of most famous web hosting startup in Vietnam 2019. TinoHost core business involves providing Shared, Enterprise, Cloud and Virtual Private Server plans. This assignment is my thesis of bachelor program which I am conducting at Saxion University of Applied Sciences, the Netherlands.

Through the graduation project have I acquired enormous amounts of knowledge about organizational working culture, which are beneficially far beyond what I could learn in a normal theoretical educational institution project. In short, would I like to thank TinoHost and the educational institution Saxion University for providing me this great opportunity where I have developed myself academically, professionally and socially.

I am very honoured and lucky with the encouragement and guidance from my company supervisors Mr. Anh Le, Mr. Binh Tran and Ms. Nhi Le. I also would like to express my gratitude to Mr. Tonny Lievers in being my academic supervisor and more importantly for his enthusiastic encouragements and precious instructions during my project period. He gave me in-time feedback on my report and participate in Skype and Microsoft Teams meetings in which I could present achievements and overall progress.

Furthermore, my gratitude to the employees of TinoHost for being supportive and sparing the time to share their knowledge in their various fields of specializations. Additionally, would I like to sincerely thank my fellow trainees for creating interesting teamwork and educational adventure.

SUMMARY

The dynamic evolvement of the Information Technology world has significant impact on other domains. All types of organizations have to keep pace with this, since the customers have less and less patience and higher demands. Therefore, have hosting services to be improved qualitatively and automatized to become more scalable and provide high-availability services, improve the employees' motivation and increase the customer satisfaction. Therefore, the Proxmox cloud solution has been researched in order to implement into a product in the near future. Because of limitation by a signed NDA (non-disclosure agreement), highly detailed information could not be described, and can only provide a surface level.

Furthermore, the report gives an insight into using learned knowledge about customer support provided. As well as implementing Research Design framework in the project. After knowing the scenario in terms of the company current situation, some of recommendation came up. The report indicates the work conducted during the project until the given moment at TinoHost. As well as consists recommendations and conclusion according to my point of view, which I assume the solution would be suitable the company if implemented correctly.

Several extra activities conducted are also described in this report. This includes the work on data centre to install the cloud server. This work requires fundamental knowledge about networking and IT infrastructure. The design of the cloud cluster was researched carefully and audited by me in collaboration with the company supervisor.

1. INTRODUCTION

This report is the result of a graduation assignment conducted within the TinoHost and is necessary as being a requirement to complete the HBO-ICT program from Saxion University of Applied Sciences. Additionally, includes this report information on the products and services of TinoHost, while providing an overview of the organization. Furthermore, the report will provide an overview into my learning objectives and assignment and how they have been achieved throughout this graduation project.

The personal learning objectives have been described in chapter three, within different goals to be achieved. Not only have the objectives to be achieved as much as possible, but also is there is a research design assignment required to be fulfilled throughout my graduation project.

2. ASSIGNMENT

There are many technologies for Cloud Computing with different price and solution. Depending on the purpose of use and the advantages of each technology such as easy deployment, high scalability, low price, etc.

TinoHost decided to be not depended on closed and copyrighted CC solution. Since commercial solutions are often a set of solutions with manufacturer standards such as specific APIs, format types image and private storage, etc. This led to the fact that the cloud incompatible may occurs, or not take advantage of the existing infrastructure. Additionally, projects on "open-source cloud computing" are always supported and helped by the worldwide in developing new and error-free functions. Cost is also an outstanding issue in opening a cloud network with copyright software solutions. Therefore, my assignment is to research a suitable open-source product for deploying Cloud environment.

3. DESIGN RESEARCH

For this project, a proper design research will be executed. For the project client, all the parts of the full design research report will be described in this chapter. The design research is the methodology that is going to be applied for this project and it is obligated to use this method.

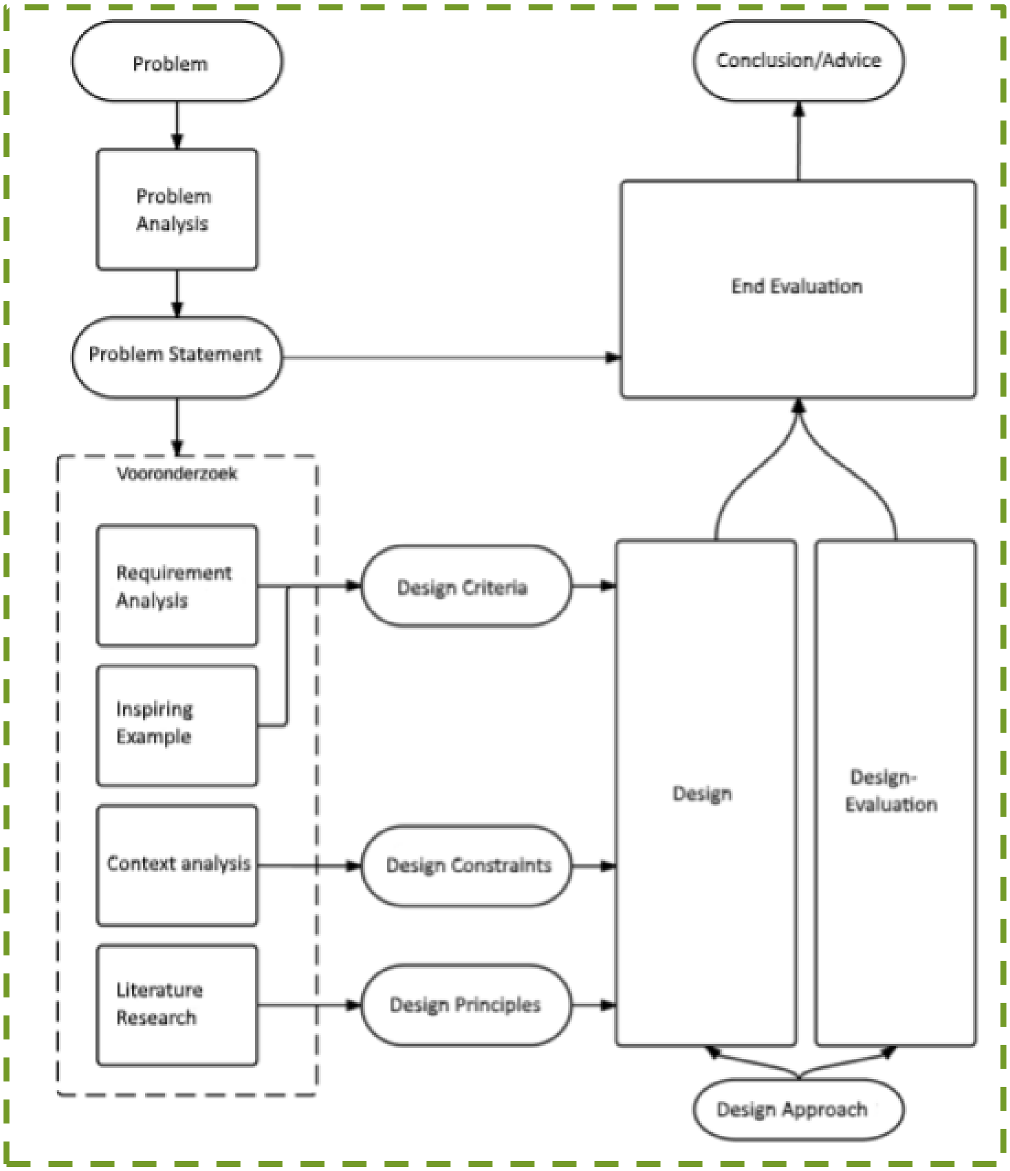


Figure 1: Design research methodology

The design research methodology is shown above. This is the methodology that is going to be used throughout this whole project. Design research is a form of research in which science-based research results in an intervention or a design for an intervention with which a concrete problem of the client can be solved [1]. It consists of different stages; these will be explained on the next pages.

3.1 Problem Analysis

The research takes times and effort; therefore, it is important to understand to know for certain that there is a problem that must be solved. The company supervisor has commissioned this project so that he can tackle a specific set of practical problems that has been identified.

Problem 1 - Researched hypervisor is maybe unstable.

Problem 2 - Require several hardware servers in order to test the hypervisor environment probably.

Problem 3 - Managing storage, data redundancy and data transport is a tough problem to solve to bring success to the project.

Problem 4 - There are also other issues that need to be posed: data security (confidentiality). For many businesses, data is the most sensitive. Giving this data to a third party is simply hard to accept. Like for example: Banks have enjoyed hiring external services for a part of their service, but want control of hardware and software - essentially wanting to use external resources just like a room, internal support staff.

Methodology

In regard to the problem analysis, the following methods will be used:

Interview: Structured, semi-structured, open ended questionnaires.

Focus group: Discussion and group interaction will produce the most useful data. It is a way to achieve proper and broad conceptualization within the project group, such as brainstorming, mind maps.

Desk research: is basically involved in collecting data from existing resources hence it is often considered a low-cost technique as compared to field research, as the main cost is involved in researcher's time, telephone charges and directories.

Triangulation: discuss at least 3 aspects of a cause or part of the problem.

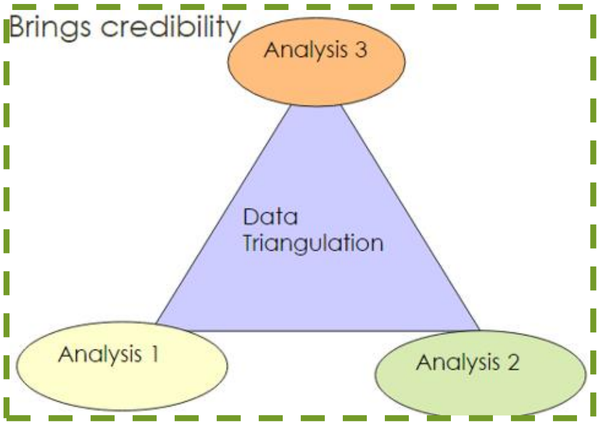


Figure 2: Data Triangulation

3.1.1 Problem analysis from different perspectives

The stakeholder’s perspective is probably not the only perspective that is important for a good and correct design. In order to make a correct problem analysis, the problems will be reviewed that the client has identified through four different perspectives: The perspective from the project client, users, literature, and personal perspective from the researcher.

The stakeholder has his own interests and these influence the nature of the problem and the solution. In the following sub-chapters analyses are done from every perspective listed above.

PROBLEM ANALYSIS FROM THE CLIENT’S PERSPECTIVE

After the initial meeting with the client, Tino Host, a slightly overview of the problem had been revealed therefore addition appointments were scheduled to have an extended interview. Mr. Le Anh, was one of the supervisors has been interviewed to retrieve more insight into their current situation design. Obviously, the problem of the client is that there is not an existing Infrastructure as a Service yet for their servers. The client requested the project team to create a small prototype version of the future design.

The problem that the researcher does care about are infrastructure related. This means that the main problem is the realization of the infrastructure for the future design. The problem statements or research question will be based on this.

The project client also pointed out that they want insight on how to build and manage high availability server cluster. Furthermore, they also would like the researcher to do research about storage solution and on how to optimize storage performance to bring the highest economic benefit.

PROBLEM ANALYSIS FROM THE USER’S PERSPECTIVE

For the cloud design, 2 different types of users can be identified be used in this design research. The first type of users, are technical staff who will administrate the system. The second group of users that are usually the end users, that will pay and use the product.

From the system administrators' point of view, the biggest concern is how to prevent and detect overload on the server when it occurs. The end users are important for Tino Host in the way that they will most likely be the revenue stream for the company. One of the issues for them is the cost, moving systems to the cloud can incur various known and surprises costs. In addition, A cloud migration means changes for the IT team. The shuffle of personnel and skills means some of current efforts are now on standby. Training up to address the slack can create delays. Hiring consultants could increase costs. Delaying internal feature development and bugfixes makes everyone unhappy.

PROBLEM ANALYSIS FROM PERSONAL PERSPECTIVE OF THE RESEARCHER

The researcher believes that the success of this project depends on how the research design will meet the needs of the project client.

3.1.2 Main problem statement

The problem analysis that was executed will act as the input to formulate the problem statement (main research question) along with sub-questions to give a better solution to the main problem statement. These research questions will be used throughout the whole design research that will be executed by the student to come up with a good solution for the problem at the design phase.

Problem statement: **How do we choose the best fit open source Infrastructure as the Service (IaaS) solution for building and managing clouds?**

SUB-QUESTIONS

1. What different kind of Hypervisor are there and what is the best practice solution for Tino Host?
2. What different kind of storage solutions are there and what is the best practice solution for Tino Host?
3. How will high availability with an uptime of 98% be realized?
4. How do we scalable storage solution for the prototype?
5. What are the different ways of doing a proper data backup and/or snapshot to saving the data?
6. What security measures are necessary for the design with regards to the networking?

3.2 Need Analysis

This analysis discusses the needs of the most important stakeholders in the project. It is important for all possible views to be considered before proceeding.

Methodology

In order to defined the important stakeholders, brainstorming method and the information got from the first meeting with the client is used.

The following methods were used to gather information about needs and also from the client interviews to compile a full list of all possible stakeholders and their needs.

Stakeholder analysis.

Interview.

Moscow and SMART methods for requirements.

3.2.1 Stakeholders analysis

As can be seen below, the extended stakeholders list and a selection of the most important stakeholders in the project. The action of this analysis will be based on different sources like interviews and brainstorming.

All stakeholders

A stakeholder is someone that have an influence within the project either positive or negative. In this section multiple stakeholders will be mentioned alongside their specific role within the project.

Stakeholder	Role with in the project
Anh Le, IT manager	The company supervisor, the commissioner of the project and supervise the project process.
Binh Tran, Project sponsor	The company director which is financing the project. They make sure the project has the financial support if needed to be completed.
Resource managers	Other managers who control the resources needed to complete the project. (a hardware server, a monitor, etc.)
System administrators	The people who will administrate the final design when its fully implemented
Customers or Users	The people who will be directly using the result of the project.

Table 1: Stakeholders

Important stakeholders

The following stakeholders have been chosen as the focus on this research since I believed that these are the stakeholders that will be most involved with the project and play an important role for its realization. There are three important stockholders in the project that are: the company supervisor, system administrators, and external customer/ end users

3.2.2 Interviews

To figure out the wishes of the stakeholder and customer, multiple interviews need to be held. These are semi-structured. This is a pre-formed questionnaire but is allowed to bend off of this. Selection of different subjects is being determined by the role as stakeholder within the project. The interview will go into the needs of the stakeholders and ideas about a possible solution. In this need analysis requirements will be made and these are part of the conclusion of this subject. Requirements are defined with the SMART method (Specific, Measurable, Acceptable, Realistic and time specific). These wishes are being define within three different categories of requirements such as: Business requirement, user requirement, and system requirement.

All requirements are important, but they are prioritized to deliver the greatest and most immediate business benefits early. Selecting different priorities is part of the requirement table. This is being done with the Moscow method.

Must have: Requirements labelled as Must have are requirements absolutely necessary.

Should have: Major importance within the solution but could get over.

Could have: Minor importance and possible to be implemented later.

Won’t have: No important for now but for a future possibility.

3.2.3 Requirements

After having multiple meetings with the stakeholders, we can mention the most important demands and wishes in the order to success the project. Currently this will be mentioned as wishes and later in the conclusion will be written as smart requirements.

The company supervisor requirements

1. Open source software.
2. A prototype.
3. High Availability (99.9% uptime, that 0.1% downtime equals about 45 minutes per month or approximately 8 hours per year).
4. A storage solution that can scale up, low at cost and that integrates well with other components.
5. IOPS limitation/ calculation for each VPS in the server to prevent overload on in the server.
6. Overload detection and prevention for the design system.

System admin requirements

1. A dashboard to monitor which VPS is overload, appropriating hard disk resources or network.
2. Backup solutions to prevent losing data and/or a snapshot solution which require less storage space and the restoration speed will be significantly increased.
3. Hot-swappable components that allow technical staff to install or remove while the system is running, without affecting the rest of the system.
4. Applying update without reboot the server.

End user requirement

1. End user features (restart, stop, reboot, recursion and capture snapshot the virtual machine. Statistic CPU, Ram and network).

RESULTS

As a final result of the need analysis the requirements are being written down and defined of priority according to the methodology.

ID	Requirement	Moscow priority
R1	Open source software	*Must*
R2	High Availability 99.9% uptime	*Must*
R3	A storage solution that can scale up, low at cost and that integrates well with other components. .	*Must*
R4	IOPS limitation/ calculation for each VPS in the server to prevent overload on in the server.	*Must*
R5	Backup solutions to prevent losing data and/or a snapshot solution which require less storage space and the restoration speed will be significantly increased.	*Must*
R6	End user features	*Must*
R7	A prototype	*Must*
R8	Hot-swappable components that allow technical staff to install or remove while the system is running, without affecting the rest of the system.	*Should*
R9	Applying update without reboot.	*Should*
R10	Overload detection and prevention in the system.	*Could*
R11	A dashboard to monitor which VPS is overload, appropriating hard disk resources or network.	*Could*

Table 2: Requirements

3.3 Context Analysis

It is essential to understand the context of the current situation, because the final design will have to fit within that context. The limitations from this specific context are a challenge for this assignment. Examples of these challenging constraints can be financial or technical limitations. Therefore, the objective of context analysis is to map out constraints from the context in a systematic way. [1]

Regularly updating the context analysis throughout the course of the project will help ensure the project can identify and adapt to changes as needed.

Methodology

The following methods were used to gather data and form the constraints:

Interviews with the client and other stakeholders.

Focus groups, discussions and group interactions such as brainstorming and mind maps.

Desk research for collecting data such as literature.

3.3.1 Constraints

This section specifies the constraint imposed on the project. These are grouped by aggregation levels such as: client, target users, project team and technology. By looking at the contexts of multiple aggregation levels we can create a better picture of all possible contextual constraints in the project. As constraints restrict project options and can result in severe trade-offs, they demand careful consideration. In most cases, constraints are driven by absolute business necessity. The following is a list of common project constraints

Client constraints

Dada centre is placed in a third-party provider, its take time create ticket for a visit and travel to there. An important constraint from the client-side is the lack of IT specialists in the organisation. Furthermore, an automated infrastructure is desired since it will save a lot of time on manual administration.

User constraints

A part of end users not feeling confident about the security of the cloud, thus deciding not to use it.

Project team constraints

Staffing constraints such as a fixed size team (only me work on this project).

Technology constraint

A limit or restriction on a facility such as physical servers for testing purpose. Since in order to create a cloud cluster there is a need of min 3 nodes and others nodes for data replication.

As with all technologies, it’s essential that administrators receive some sort of training. Since when the cluster goes live and become a business dependency, unexperienced administrator is a constraint to stability.

3.4 Literature Research

The problem, needs and context analyses have been properly executed. The research phase actually started already from the problem analysis, since doing research is an ongoing process during the project. For every research question that was formulated in the earlier stages of the design research, it is looked at possible solutions and inputs for the particular research question.

This literature research will result in design principles that will be used in the technical design for defining the final solutions. The design principles are the conclusions of the literature research – giving answers to the all the research questions.

Methodology

When searching for literature, I have gathered the data by utilizing the search tree and the snowball methods [2]. The search for literature was done mainly with Google Scholar and the Saxion library. Search strings are used which derive from search questions or sub-questions in the problem analysis report. This way I can find suitable articles of high quality which can be use as input. Furthermore, I am also using the snowball method. By using the bibliography of already found articles to find suitable references which I can also use as input.

Please refer to the search tree table in the appendix for more detailed information.

3.4.1 What is cloud computing and what is the best practice cloud solution for TinoHost cloud environment?

In this chapter, there will be a comparison about visualization technologies. First, definition of cloud computing will be discussed along with the advantages. Finally, possible hypervisors that suit the TinoHost environment will came out. In the conclusion, the recommended solution will be proposed.

Commercial solutions will not be taken into account, because of the high total cost of ownership. They are often resource intensive on both the CPU and RAM capacity. Furthermore, licensing costs can be expensive, whereas open source solutions are free and less resource intensive. Therefore, this report will only look at open source hypervisor solutions.

THE NIST DEFINITION OF CLOUD COMPUTING

Cloud computing can be simply understood as an information system where the provider has built the necessary services. From there, the users can rent to use the necessary services and will only pay for those services. This save a lot of time, effort, cost, and manpower to build an IT infrastructure on premise. This cloud model is composed of five essential characteristics, three service models, and four deployment models. [3]

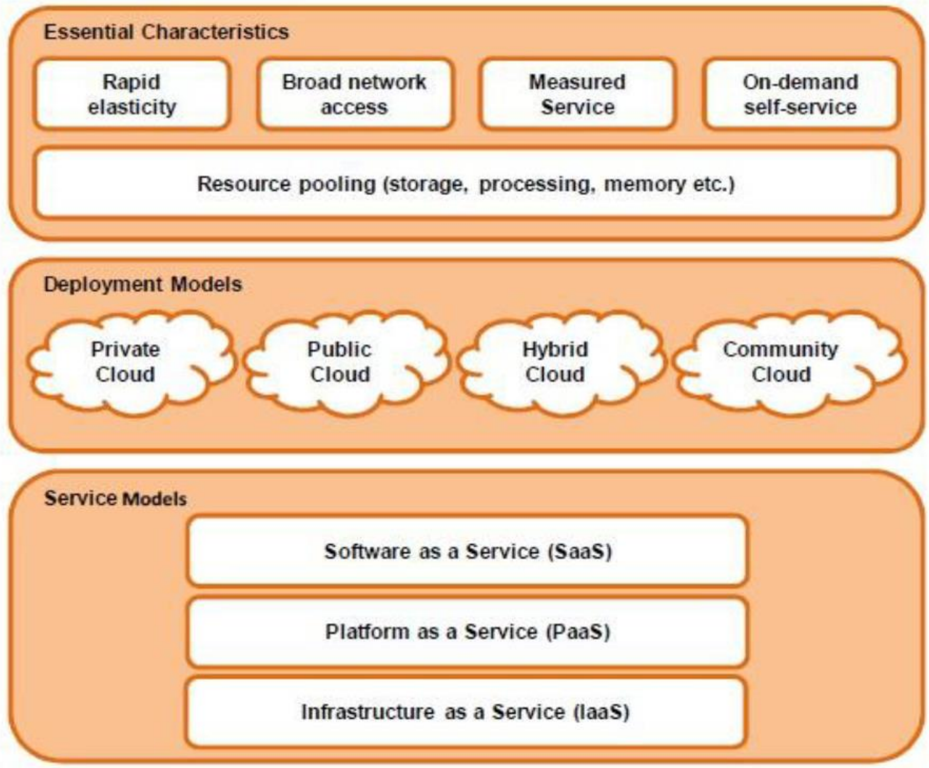

Essential Characteristics
Rapid elasticity
Broad network access
Measured Service
On-demand self-service
Resource pooling (storage, processing, memory etc.)
Deployment Models
Private Cloud
Public Cloud
Hybrid Cloud
Community Cloud
Service Models
Software as a Service (SaaS)
Platform as a Service (PaaS)
Infrastructure as a Service (IaaS)

Figure 3: cloud computing definition (NIST)

Five essential features in CC may vary depending on the actual deployment model. For example, in the private cloud model, the resources used by only one enterprise. The "On-demand self-service" or "resource pool" feature will be different from other models.

Rapid elasticity: CC provider easily assigns and retrieves user resources very quickly. The user is allowed to request an unlimited resource and of course they have to pay for that.

Broad network access: access to computer resources easily through standard network mechanisms (e.g. mobile phone, tablet, notebook).

Measured service: provider ensures the calculation of customer consumption. The target model is "pay as you go". Similar to Amazon Web Service (AWS) and Google Cloud Platform (GCP), They are billing based on the resource used by cloud users.

On-demand self-service: customers are allowed to customize resource usage without notice or through any provider intervention.

Resource pooling: CC's physical and virtual resources (storage, processing, memory, and network bandwidth) are shared with each other and automatically allocated to users.

There are three main models of cloud computing: public, private, and hybrid (hybrid between public and private cloud). The public cloud is a cloud model on which cloud providers provide services such as resources, platforms, or external and cloud based public storage applications. Services on the public cloud can be free or paid. Private cloud services are provided internally and often business services, aimed at providing services to a group of people and it is behind a firewall. A hybrid cloud is a cloud environment that incorporates private and public services. There is also a community cloud that is a cloud among cloud service providers.

Based on a service the cloud model is offering, we are speaking of either: SaaS (Software as a Service), PaaS (Platform as a Service), IaaS (Infrastructure as a Service) [4].

Before going into each type of service, we will look at the image depicted below to see where each type of service is in the cloud architecture model.

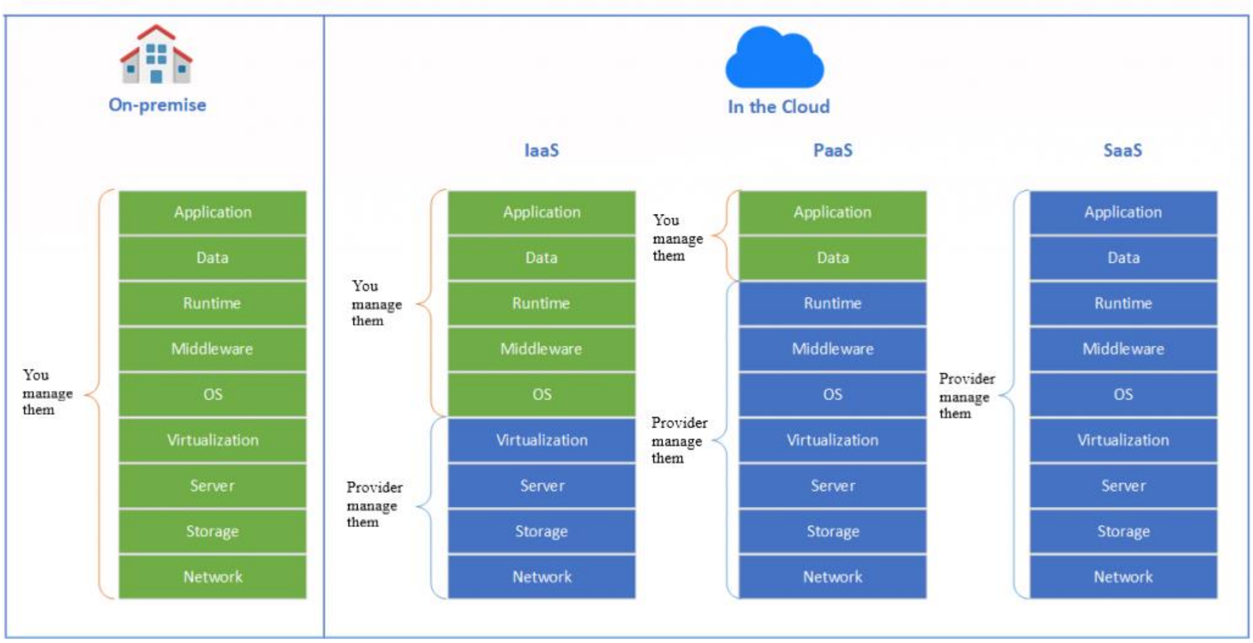


Figure 4: cloud architecture model [5]

In On-premise model (build your own IT system), the individual business will have to manage a system built by themself. Meanwhile, in the cloud model, businesses will only hire necessary services without worry about the lower layers (network, storage, server, etc). Depending on the needs, businesses and organizations will choose different services, corresponding to SaaS, PaaS, IaaS.

SaaS – Software as a Service

Software that is available via a third-party over the internet. The software is running on a cloud infrastructure and accessible from various client device through client interface, such as a web browser. The users are not responsible for hardware or software updates.

It is clear that SaaS would bring about several advantages. One evident strength is that it would reduce time for installing software, managing and upgrading software. Another reason is that it would help enterprises focus more on business than handling issues, system administration.

SaaS is suitable for small startups, need to run website services for quick marketing or short-term projects that require collaborators to work remotely. There are some well-known SaaS on the market like, for example: G Suite (Google), Dropbox, MailChimp, Concur, Salesforce, Cisco WebEx, Slack, etc.

PaaS – Platform as a Service

PaaS provides an environment for developers that they can create, deploy or test applications by use programming languages, libraries, services, and tools supported by the provider. The user does not have to care about the network, storage, server, and operating systems. Since PaaS built on virtualization technology, meaning that resources can easily be increased or decreased when demand changes.

There are some obvious advantages that would arise. First, PaaS is flexible for libraries and frameworks. In particular, it supports many different languages: Nodejs, Java, Ruby, C#, Python, PHP depending on cloud solution's provider. Second, it can be easily extended and multiple users can access the same application developed.

PaaS is suitable for technology startups that are in need of a quick build and scaling system. Or software company need to develop applications quickly and easily. Some example of PaaS: Google App Engine, Elastic Beanstalk - Amazon, Cloud Services - Azure, Openshift, etc.

IaaS – Infrastructure as a Service

IaaS provider offer services such as pay-as-you-go storage, networking, and virtualization. IaaS gives users cloud-based alternatives to on-premise infrastructure, so businesses can avoid investing in expensive on-site resources. Unlike SaaS or PaaS, customers will be responsible for managing their own applications, data, runtime, middleware, and operating systems.

The advantages of IaaS solution would prove to be worthwhile. First, it is highly flexible, scalable and cost-effective. In fact, users only have to pay for what they used and easy to scale the system when their need is increased. In addition, PaaS is accessible by multiple users and give them full control of the infrastructure.

This model is suitable for startups or small companies, IaaS is an option so they don't have to spend time for hardware. Moreover, IaaS is also beneficial for large organizations who want full control over their applications and infrastructure. Example of leading IaaS providers: Google Compute Engine (GCE), AWS EC2, Azure Virtual Machine, Linode, Digital Ocean.

Overall, we can encapsulate the services in the following model:

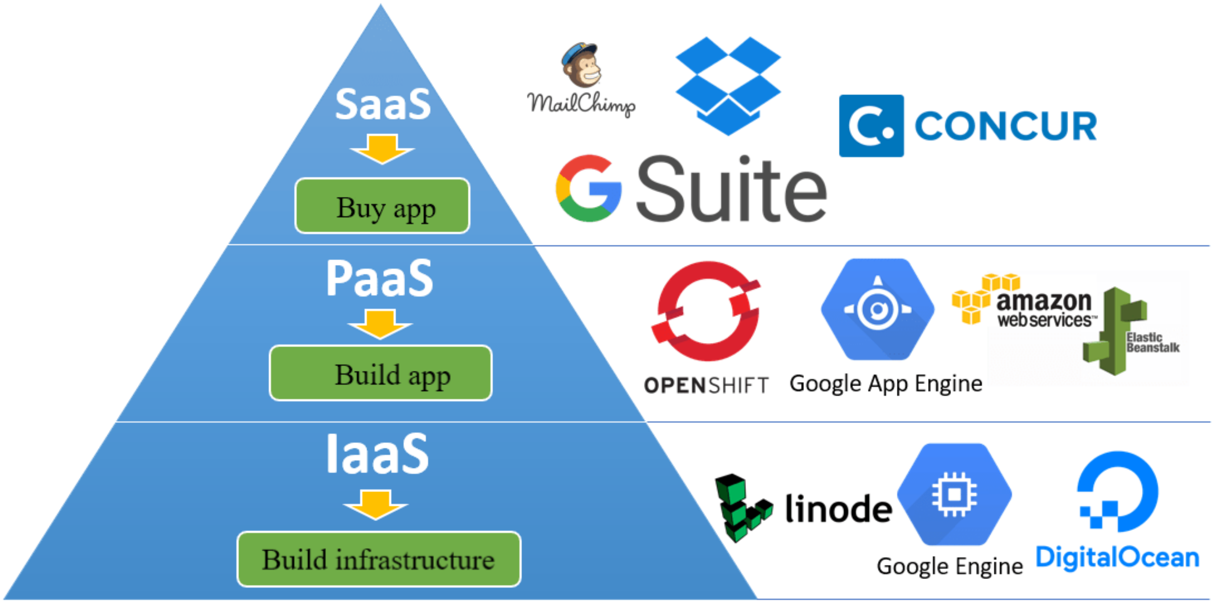


Figure 5: SaaS, PaaS, IaaS services model [5]

THE BENEFITS OF CLOUD COMPUTING

There are some benefits and features of the cloud system:

Increased Flexibility: when it is necessary to add or remove one or several devices (storage devices, servers, computers, etc) just takes a few seconds.

IT resources on demand: depending on the needs of the customer, the administrator configures the system to provide customers.

Increased availability: Increased availability of applications and services to ensure availability. When one of the hardware failures, it does not affect the system, only degrades system resources.

Hardware saving: In most cases, the traditional model needs a separate system for each task and service. By contrast, in the Cloud Computing model, IT resources are managed to ensure this wastefulness.

Paying as you go IT: Cloud computing model is integrated with a billing system to perform billing based on used resources of the user, such as CPU speed, RAM capacity, HDD capacity, etc.

In summary, the CC model has overcome two important weakness of the traditional model that is scalability and flexibility. Organizations and companies could deploy applications and services quickly, at low cost, and with little risk. The next section will introduce virtualization, the core technology and seen as a transition step from the traditional model to CC model.

VIRTUALIZATION TECHNOLOGIES

All types of virtualization are managed by VMM (Virtual Machine Monitor). Hypervisor or VMM is divided into two types: Type 1 hypervisors run directly on the system hardware and thus are referred to as bare-metal hypervisors; Type 2 hypervisors run on the operating system to share resources with the operating system and are referred to as hosted

Example for type 1 hypervisors: VMware ESXi (monolithic architecture), Microsoft Hyper-V (Microkernelized design), Citrix XenServer (Microkernelized design), Oracle VM, ProxmoxVE, KVM (type 1 or type 2 still up for discussion).

Example for type 2 hypervisors: VMware Workstation/Fusion/Player, Oracle VirtualBox, Microsoft Virtual PC. QEMU, KVM.

Since it has direct access to the hardware resources rather than going through an operating system, a bare-metal hypervisor is more efficient than a hosted architecture and delivers greater scalability, robustness and performance [6].

Moreover, (Type 1) bare-metal hypervisors can be further classification into two subcategories: Monolithic and Microkernelized designs. The difference between them is the way of dealing with the device drivers.

Monolithic – Having all device drivers reside within the hypervisor. VMs access system resources through hypervisor drivers. The biggest advantage of this design is that it does not need a host operating system. This is high performance, but when the driver on the hypervisor fails, the whole system stops working, or faces security vulnerability when the driver can be disguised by malware.

Microkernelized – This type of hypervisor does not have a driver inside the hypervisor but runs directly on each partition. Virtualization stack and hardware specific device drivers are located in a specialized partition called the parent partition. The parent VM also includes many other features such as memory management, driver storage, etc. This provides safety and reliability. However, it also encountered an availability problem when the parent partition had a problem and the system stopped.

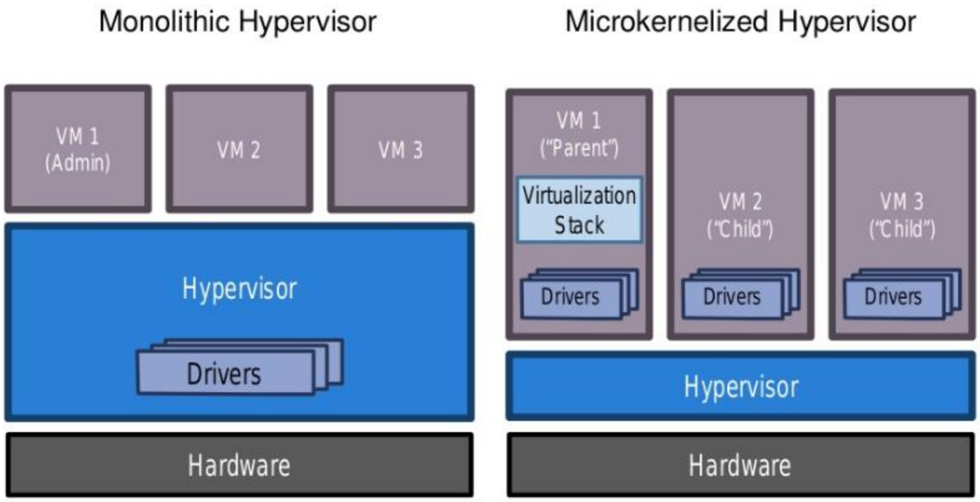


Figure 6: Monolithic hypervisor vs. Micro kernelized hypervisor [7]

Virtualization approach

Full-virtualization is designed to provide completely abstraction of the underlying hardware and creates a complete virtual system in which guest OS unaware that it is a guest. However, this virtualization approach cannot exploit its performance effectively through a hypervisor or VMM to interact with system resources. Therefore, some features will be restricted when needed directly from the CPU. Xen, VMWare workstation, Virtual Box, Qemu/KVM, and Microsoft Virtual Server support this type of virtualization. [8]

In contrast, Para-virtualization, or "partial" virtualization, is a virtualization technique supported and controlled by a hypervisor, but guest OS does not execute commands through the hypervisor. Therefore, there is no restriction on permissions. The downside of this type of virtualization, however, is that OSs know they are running on a virtual hardware platform and are difficult to configure settings. Para-Virtualization virtualization is supported by Xen, VMware, Hyper-V, and UML. [9]

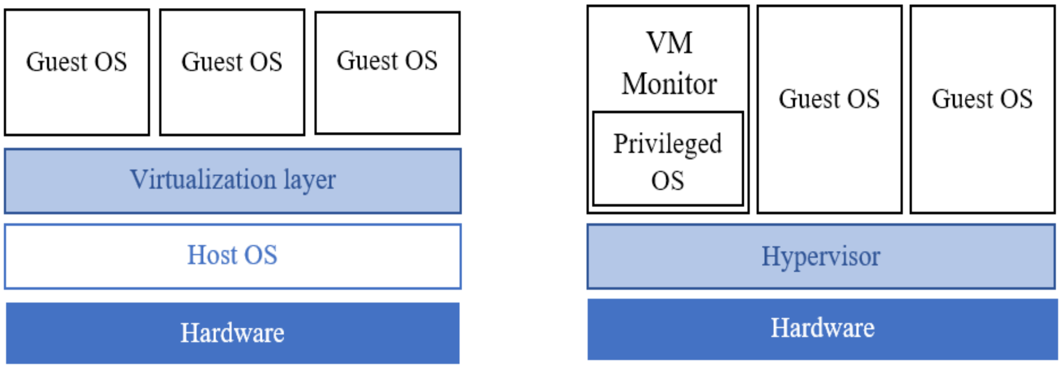


Figure 7: Full virtualization (left) vs para-virtualization (right)

There is a new virtualization method that is OS level virtualization, also known as Containers Virtualization or Isolation. This virtualization method that allows the kernel of the operating system to support multiple isolated instances based on an operating system available for different users, or in other words create and run many isolated and secure virtual machines that share the same operating system. The advantage of this virtualization is fast maintenance and it is widely used in virtual hosting environments, where it is useful for securely allocating finite hardware resources among a large number of mutually-distrusting users. Virtuozzo, Proxmox, Linux-VServer, Solaris Zones, and FreeBSD Jails support this type of virtualization [10]. One note is that this type of Containers virtualization only exists on Linux operating systems.

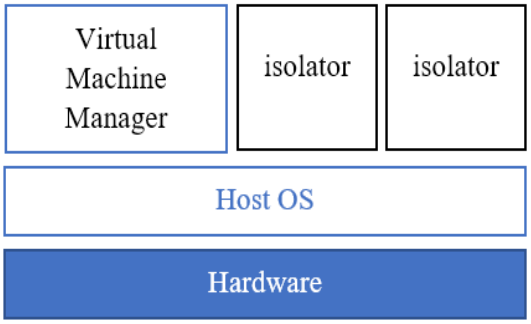


Figure 8: Isolation virtualization

Containers are a great way to bundle and run applications. In the production environment, the containers that run application have to be managed and guaranteed that there is no downtime. For example, in case a container goes down, another container need to starts. It would be much easier to have a system that can handle this behaviour and this is how Kubernetes come to the rescue.

Kubernetes, or k8s, is an open source platform that automates Linux container operations. It eliminates many manual processes involved in deploying and expanding containerized applications. Kubernetes provides a framework for running powerful distributed systems. It takes care of scaling and failover for production application, providing deployment templates and more. It was originally designed by Google and is now maintained by the Cloud Native Computing Foundation.

In general, virtualization is a foundation element of CC, the actual implementation of CC is based on two basic solutions: using commercial products for CC such as VMware, Microsoft (Hyper-V), or Open source products like OpenStack and Proxmox. The following section discusses the benefits of an open source CC deployment approach.

EXISTING SOLUTION/ PRODUCTS

VMware vSphere, Microsoft Hyper-V, Citrix XenServer, Oracle VM and Red Hat KVM are the biggest players in the server virtualization market. A comparison between the best server virtualization software on the basis of features and hardware requirements will make it easier to select the best hypervisor for TinoHost.

VMWARE ESX AND ESXI

These hypervisors offer advanced features and scalability but require licensing, so the costs are higher. There are some lower-cost bundles that VMware offers and they can make hypervisor technology more affordable for small infrastructures. VMware is the leader in the Type-1 hypervisors. Their vSphere/ESXi product is available in a free edition and 5 commercial editions.

The main advantage of ESXi is flexibility and ease to use. In fact, it has a user-friendly interface, with simple installation and usage. ESXi supports multiple operating systems, various versions, and advanced features. Another additional advantage is that each virtual machine has a different IP. Therefore, VMware can run and work independently with other VMs. As well as working with many types of operating systems (e.g. Windows, Linux), and it can run and use any specific software designed for different operating systems. That makes VMware be a great tool to help to test software on all operating systems before delivering it to end user.

However, ESXi is not completely beneficial, there are several negative aspects that should be taken into consideration. First, ESXi is highly cost. It means that it is not a choice for small businesses at a low cost. Secondly, ESXi has far more hardware requirements than Proxmox. In particular, VMware is owned by Dell, a hardware manufacture. Each release of VMware stops supporting some older hardware. The minimum requirement to install is also high, it is recommended to not run the ESXi server with less than 8GB RAM. [11]

MICROSOFT HYPER-V

Although Hyper-V doesn’t offer many of the advanced features that VMware’s products provide. However, with XenServer and vSphere, Hyper-V is one of the top 3 Type-1 hypervisors on the market. Hyper-V is available in both a free edition (with no GUI and no virtualization rights) and 4 commercial editions - Foundations, Essentials, Standard, and Datacenter. Microsoft Azure cloud system use a hypervisor which is a customized version of Hyper-V, known as the Microsoft Azure Hypervisor to provide virtualization of services.

CITRIX XENSERVER

It began as an open source project. The core hypervisor technology is free, but like VMware’s free ESXi, it has almost no advanced features. Xen is a type-1 bare-metal hypervisor. Just as Red Hat Enterprise Virtualization uses KVM, Citrix uses Xen in the commercial XenServer. Nowadays, XenServer is a commercial type-1 hypervisor solution from Citrix, offered in 4 editions.

ORACLE VM

The Oracle hypervisor is based on the open source Xen. However, it will cost money for the hypervisor support and product updates. Oracle VM lacks many of the advanced features found in other bare-metal virtualization hypervisors.

KVM

KVM (Kernel-based Virtual Machine) initially developed by Qumranet later acquired by Redhat in 2008. It is an open source virtualization technology built into Linux. KVM turns Linux kernel into a hypervisor and allows host machine to run multiple, isolated VMs or guests.

It is also believed that VMware has its own benefits and drawbacks. One of the important benefits is flexibility. Although the server is installed with Linux, KVM supports creating VMs that can run both Linux and Windows. Used in combination with QEMU, KVM can run Mac OS X. KVM also supports both x86 and x86-64 systems.

Another strong point of KVM is cost savings and high scalability. In other words, KVM is developed on a completely free and open source platform, supported by the community and from device manufacturers. It is growing and becoming one of the best choices for low-cost small businesses.

Despite these attractions, however, some drawbacks do exist. The main disadvantage of KVM is it requires a high configuration server. As a full virtualization technology, KVM requires an advance setup server. Even requiring servers of reputable brands like IBM and Dell to ensure good operation.

CLOUD COMPUTING APPROACH USING OPEN SOURCE PRODUCTS

With the benefits of CC mentioned in the previous chapter. There are many technologies for CC with different costs and solutions, which is depending on the purpose and advantages of each technology such as: easy to deploy, high scalability, and low price. Using open source tools to deploy CC achieves the following advantages: [12]

> *Avoid vendor locking*: Commercial solutions are usually a set of solutions with manufacturer's standards such as specific APIs, image formats and storage, etc. will make the cloud incompatible, or cannot take advantage of the available infrastructure. Or in the future, vendor lock-in clouds will face the problem when migrating some services to other cloud systems.
>
> *Getting best-of-breed technology*: Open source CC projects are always supported by the worldwide community. With thousands of people involved in developing new functions and fixing bugs, this advantage will only have open source products.
>
> *Scalability without restrictions*: License cost is a prominent issue while expanding the cloud. However, with open source clouds, for example, the cloud system uses Ubuntu. Ubuntu operating system supports CC completely free, so it is easy to expand.
>
> *Aligning the cloud to specific business needs*: When a commercial solution lacks a function, it is difficult to find an alternative unless waiting for a newer version to support it. But with open source technology, it is possible to change the code to add functions suitable for the business purposes of the system.

There are some of the open source cloud solutions such as: OpenStack, Eucalyptus, and Proxmox.

OPENSTACK

OpenStack is a community open software for developing CC suitable for vendors (Cloud Providers) as well as users (Cloud Customers) developed by Rackspace hosting and Nasa. OpenStack has a modular design, meaning it includes many subprojects, each project plays a specific role and is related to each other. OpenStack consist of three main projects: OpenStack Compute (to deploy the management and allocation of resources for virtual instances), OpenStack Object Storage (execute storage, backup), and OpenStack Image Service (undertake the delivery. displaying, registering, transmitting services to virtual disk images).

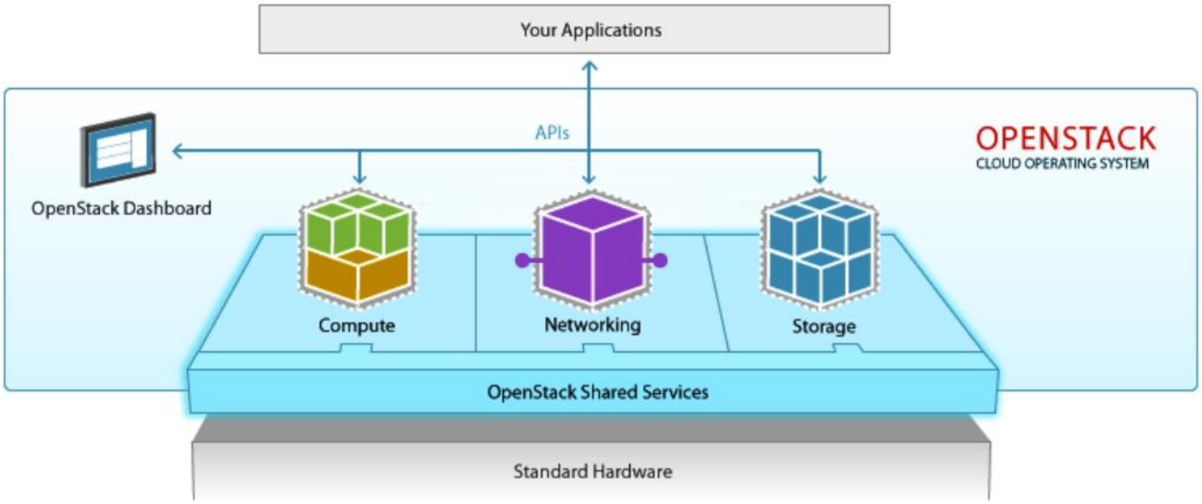


Figure 9: OpenStack architecture [13]

OpenStack compute (Nova) **-** This is the most basic part of OpenStack that controls IaaS and redistributes system resources to instances with independent storage computing capabilities. Typically, Nova gives users the ability to run instances (virtual machines) and the interface to manage those instances on the hardware infrastructure. However, Nova does not include any virtualization software. What it does is reuse hypervisors (optionally installed by the user) to perform virtualization calculations. Users can use different hypervisors in different zones. Here are the hypervisors that Nova currently supports: Hyper-V, KVM, LXC. QEMU, UML, VMWare ESXi, XEN.

The figure below shows the number of OpenStack Nova drivers for Hypervisors, which allow administrator to choose which Hypervisor(s) to use for the Nova deployment. The drivers are divided into three groups A, B, C, corresponding to the decreasing test level.

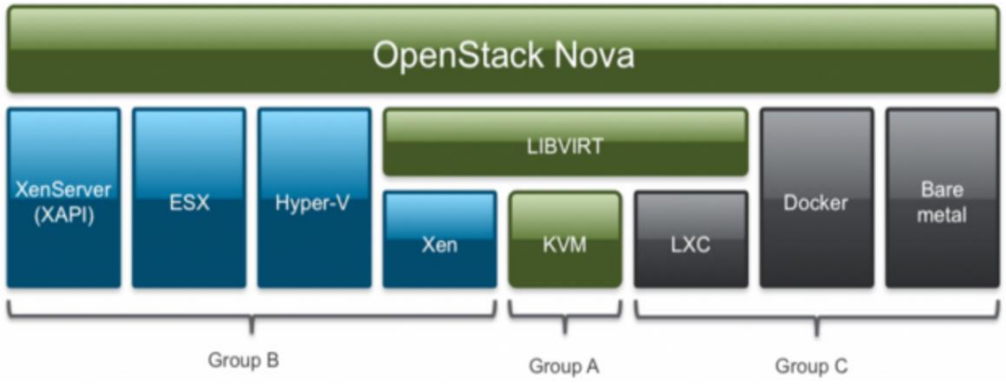


Figure 10: Nova supported hypervisors [14]

An important factor in the choice of hypervisor is depending on organization’s hypervisor usage or experience. Moreover, hypervisor’s feature parity, documentation, and the level of community experience are also important factor. According to recent OpenStack user survey, KVM is the most widely selected hypervisor in the OpenStack community. Besides KVM, there are many deployments that run other hypervisor such as LXC, VMware, Xen, Hyper-V. However, these hypervisors are either less utilized, are specialty hypervisors, or have restricted usefulness compared to more commonly used hypervisors. [13]

OpenStack Object Storage (Swift) - OpenStack Object Storage, also known as Swift, has been open-source by Rackspace since 2010. It is the technology used behind Rackspace's Cloud Files, one of the good commercial storage solutions, which is currently competing with Amazon S3. In OpenStack, Swift play the role of an object storage to stores the image OS of VM.

OpenStack Image Service (Glance) - Glance manages the VM image OS information but the images themselves are stored on Swift. Glance is equipped with APIs to get these images for Nova**.** To describe the function of Glance, we can simply describe by the operation diagram as follows:

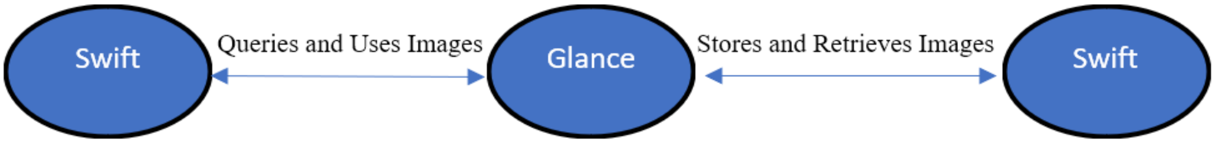


Figure 11: Glance operation diagram

To sum up, OpenStack is being evaluated as the most powerful open source software for CC construction with the support of major computer manufacturers in the world such as HP, Canonical, IBM, Cisco, Microsoft, etc. This is also a toolkit. Important issues are underway and will be detailed in the following sections. Overall, OpenStack seems like a better framework for the future if we want to expand. However, it appears too complex just for two Tinohost employees to manage time to build the infrastructure.

EUCALYPTUS

Eucalyptus is an open source Linux-based software for deploying CC with both private and hybrid (private and public). Eucalyptus provides IaaS (Infrastructure as a Service) to facilitate the allocation of resources (hardware, storage, and network infrastructure) based on usage requirements. The strength of Eucalyptus is to deploy enterprise data centres without too many hardware configuration requirements. Moreover, Eucalyptus supports connecting with famous Amazon cloud service - AWS (Amazon Web Services) through a common programming interface. The architecture of Eucalyptus is simple, flexible (flexible), modularized (Modular) and has many advantages such as snapshot, self-service, and so on.

PROXMOX VE

Proxmox VE is a complete open source platform for enterprise virtualization including tight integration of KVM hypervisor and LXC containers, software-defined storage and network functions on a single platform, and easy management. High availability clusters and disaster recovery tools in the web management interface.

It is clear that Proxmox VE would bring about several benefits in terms of the value aspect. One evident strength is that It is a completely opensource product, which makes Proxmox is a clear winner on value. For example, VMware requires vendor representatives to take “VCP” certification course, which costs $4,250. Another benefit is that Proxmox features are practical. For example, Proxmox automatically allows nodes to use the same shared storage when the user adds them to a cluster. In addition, Proxmox ease to use with the graphical interface. It means that Proxmox is easy to use after installation. Because the base OS for Proxmox is Debian Linux. Therefore, administrators can apply their existing Linux knowledge to use Proxmox CLI. Furthermore, Proxmox’s hardware requirements are much simpler only requiring CPU that supports ether AMD-V Or VT-x.

CONCLUSION

In conclusion, Proxmox VE is recommended for the choice of CC system. Since, VMware ESXi, Microsoft Hyper-V cost the license and the output bandwidth. Therefore, this is the reason why we do not choose a commercial product. There are some reasons that we do not choose another popular opensource solution such as OpenStack. Firstly, OpenStack is too complicated for our environment, then the most likely we enjoy this hypervisor as it is aimed for small deployments. Secondly, it is complicated when an update is needed since OpenStack is built from many selected modules. Therefore, when update we have to create test cases in order to test each module before actually updated. Last but not least, brand names also affect more or less the choice of the user, they are easily attracted to famous names such as OpenStack.

3.4.2 What different kind of storage solutions are there and what is the best practice solution for TinoHost cloud environment?

One of the most problematic yet important components of designing cloud infrastructure is storage. In this project, the TinoHost cloud environment needs an opensource storage that can scale up, cost-effective and integrate well with Proxmox.

There are different types of storage system with many different features, performances, and use case scenarios. The main responsibility of a storage is to hold virtual disk images, ISO or container templates, backups and so on. Therefore, it is crucial to have a proper understanding of different types of storage in order to provide the right type of storage for the right scenario. Whether our solution is a local storage configured with direct attached disks or a shared storage with hundreds of disks. In this chapter we will find out a storage solution that integrate well with the Proxmox cluster.

STORAGE TYPES

Directory

The directory storage is mainly used as local storage. VMs should not be stored in this Director because is not allow live migration. All virtual disk image file type can be stored in the Directory storage, for example ISO and container templates.

iSCSI

Internet Small Computer Systems Interface stands for iSCSI, which allows the transmission of SCSI commands over a standard IP-based network. Therefore, ISCSI takes advantage of TCP / IP and Ethernet. Especially effective when used with Ethernet 10G. In order to store virtual disk image, we must configure LVM storage on top of the iSCSI devices and then store the disk image.

Content types	Image formats	Shared	Snapshots	Clones
images none	raw	yes	no	no

Table 3: Storage features for backend iSCSI

Logical Volume Management

Logical Volume Management (LVM) is a light software layer on top of hard disks and partitions. It can be used to split available disk space into smaller logical volumes. LVM is widely used on Linux and makes managing hard drives easier.

Another use case is to put LVM on top of a big iSCSI LUN. That way you can easily manage space on that iSCSI LUN, which would not be possible otherwise, because the iSCSI specification does not define a management interface for space allocation.

LVM is a typical block storage, but this backend does not support snapshot and clones. Unfortunately, normal LVM snapshots are quite inefficient, because they interfere all writes on the whole volume group during snapshot time. The newer LVM-thin backend allows snapshot and clones, but does not support shared storage.

Content types	Image formats	Shared	Snapshots	Clones
images rootdir	raw	possible	no	no

Table 4: Storage features for backend LVM

NFS

Network File System (NFS) backend is based on the directory backend, so it shares most properties. NFS is easy to set up and require the least hardware cost. Hence, allowing a budget-conscious small business to implement on a stable shared storage system for the Proxmox cluster. NFS does not support snapshot, but the backend uses qcow2 features to implement snapshots and cloning. FreeNAS can be use as the NFS server, we can use its features and GUI to easily monitor the shared storage.

Content types	Image formats	Shared	Snapshots	Clones
images rootdir vztmpl iso backup snippets	raw qcow2 vmdk	yes	qcow2	qcow2

Table 5: Storage features for backend NFS

ZFS

ZFS storage is a combination of file system and LVM, which is a high capacity storage with important features such as data protection, data compression self-healing and snapshots. ZFS has built-in software-defined RAID, the ZFS pool support the following RAID types:

RAID-0 pool: Requires at least one disk.

RAID-1 pool: Requires at least two disks.

RAID-10 pool: Requires at least four disks.

RAIDZ-1 pool: Requires at least three disks.

*RAIDZ-2 poo*l: Requires at least four disks.

Content types	Image formats	Shared	Snapshots	Clones
images rootdir	raw subvol	no	yes	yes

Table 6: Storage features for backend ZFS

ZFS pool only function locally, nodes in the Proxmox cluster will not be able to share the storage. By mounting the ZFS locally and creating the NFS share via CLI. It is possible to share the ZFS pool between the nodes.

Ceph RBD

RADOS Block Device (RBD) storage is provided by Ceph distributed storage system. Its complex and requires multiple nodes to be set up. It provides excellent performance, reliability and scalability. In fact, Ceph can be spanned over several dozen nodes and scaled to several petabytes or more. To expand a Ceph cluster, a hard drive or a node can be simply added and Ceph will automatically rebalance data to adapt the new hard drive or node.

Content types	Image formats	Shared	Snapshots	Clones
images rootdir	raw	yes	yes	yes

Table 7: Storage features for backend RBD

GlusterFS

GlusterFS is a scalable distributed file system, which can be scaled to several petabytes. The system runs on commodity hardware and provide a high available enterprise storage at low costs. However, after a node crash, GlusterFS use “rsync” to make sure data is consistent. This take a very long time with large files therefore this backend is not suitable to store large VM images.

Content types	Image formats	Shared	Snapshots	Clones
Images vztmpl iso backup snippets	raw qcow2 vmdk	yes	qcow2	qcow2

Table 8: Storage features for backend GlusterFS

NONCOMMERCIAL VERSUS COMMERCIAL STORAGE OPTIONS

The different between non-commercial and commercial storage solution is the provider company support behind it. Consistently, non-commercial solution only has community support through forums and message boards. Commercial solution comes with technical support and, in some case, an ongoing service-level agreement (SLA) contract. The following list illustrates some non-commercial and commercial options out here to set up a storage system for the Proxmox cluster.

Non-commercial		Commercial	
napp-IT	www.napp-it.org	Nexenta	www.nexenta.com
FreeNAS	www.freenas.org	Falconstor	www.falconstor.com
GlusterFS	www.gluster.org	EMC2	www.emc.com
Ceph	www.ceph.com	Open-E DSS	www.open-e.com
NAS4Free	www.nas4free.org	NetApp	www.netapp.com

Table 9: Non-commercial/ commercial storage options

Regarding to the requirement, open-source/ non-commercial product is the only option and because it's free, there's not much choice(Ceph, GlusterFS, FreeNAS, and etc).

LOCAL STORAGE VERSUS SHARED STORAGE

In fact, shared storage is not actually necessary in a Proxmox cluster. In a small business environment, a local storage will suffice if we do not need to have 24/7 uptime and 100% reliability. However, in most virtual enterprise environments with crucial data, shared storage is the only logical choice. The following are benefits of using shared storage:

1. Live migration of virtual machine

Live migration is a virtual machine can be moved to a different node without shutting it down first. This is an important reason to go for a shared storage system. Since the hardware and operating system of Proxmox nodes sometimes need updates, security patches, and replacement. These updates require an immediate reboot, and when the node reboot, all the running VMs must be stopped or migrated to other nodes. Then, after the reboot cycle is finished VMs are migrated back to the original node.

Typically, a cluster setup with local storage can cause unwanted downtime when migration is needed. Since the VMs need to be powered off prior to migration and Proxmox using "rsysnc" to move entire image file. Therefore, migration from one local storage to another local storage takes a long time, depending to the size of VM. Whereas, if a cluster is setup with shared storage, migration do not need to power off VMs. In fact, when a node needs to be rebooted due to a security patch or update, all the virtual machines can be simply migrated to another node without powering down a single virtual machine. A virtual machine user will never notice that their machine has actually moved to a different node.

2. Seamless expansion of multinode storage space

Since shared storage is separated from the virtual machine host nodes, storage space could increase on demand without shutting down or interrupting any running critical nodes or VMs.

3. Centralized backup

By allowing VM host node to create backup in one separate backup node, shared storage make centralized backup possible. This help an administrator to implement backup plan as well as manage the existing backups.

4. Multilevel data tiering

Different file can be stored on different storage pool based on their performance needs. A virtual file server can provide very fast service if its VM is stored in an SSD storage pool, in contrast a virtual backup server could be stored in slower HHD storage pool since backup files are not usually accessed and therefore do not demand very fast I/O.

5. Central storage management

NAS, SAN, and other shared storage come with their own management program. The Ceph storage is configured via CLI, however, Proxmox had integrated Ceph management options within Proxmox GUI. This makes Ceph cluster management much easier.

The following table is a summary comparison between local and shared storage:

Features	**Local storage**	**Shared storage**
VM live migration	No	Yes
High availability	No	Yes, when use in distributed shared storage
Cost	Lower	Significantly higher
I/O performance	Native disk drive speed	Slower than native disk drive speed
Skill requirements	No special storage skills needed	Must be skill in the shared storage option used
Expandability	Limited to available drive bays of a node	Expandable over multiple nodes or racked, when use in distributed shared storage
Maintenance complexity	Virtually maintenance free	Storage nodes or clusters require regular monitoring

Table 10: Local versus shared storage multi-criteria analysis

Shared storage can cause a single point failure if a single node-based shared storage solution is set up (FreeNAS or NAS4Free without high availability configured). Thus, the better choice is using multimode or distributed shared storage such as Ceph or GlusterFS. I have compared the features of Ceph and GlusterFS in the following table.

Criteria group	Ceph	GlusterFS
License	Non-commercial	Non-commercial
Type	Distributed shared storage	Distributed shared storage
Scale-up and scale-out	Yes, easy to integrate new storage devices into an existing storage offering.	Yes, easy to integrate new storage devices into an existing storage offering.
High availability	Yes, use replication that writes data to different storage nodes simultaneously.	Yes, use replication that writes data to different storage nodes simultaneously.
Commodity hardware	Yes	Yes
Flexible	Ceph is a more flexible offering that is easier to integrate in Linux and non-Linux environments.	GlusterFS very easily into a Linux-oriented environment, integrating GlusterFS in a Windows environment is challenging
Popularity	Ceph is more popular and Ceph has been largely adopted by the open-source community, with different products available on the market.	GlusterFS is less popular.

Table 11: Ceph versus GlusterFS comparation

As the result, Ceph really does outperform GlusterFS. That’s why Ceph is recommended for the storage option to integrate with Proxmox VE.

3.4.3 What are the different ways of doing a proper backup and restore?

In this chapter, we look at the backup and restore features in Proxmox. Besides that, we also look at the VM replication feature for safekeeping when using local storage.

A good backup is the last line of defence against disaster such as hardware failure, accidental deletions, or misconfigurations. Restore feature is important as backup feature, since backup files will be nothing if the restore ability is not able in times of need. Proxmox EV provides a fully integrated solution, there are two backup options in Proxmox: full backup and snapshots.

FULL BACK UP

A full backup is a complete, compressed full backup of a VM. Different backup modes offer different data assurance and speed. There are three types of modes available for a full backup:

Snapshots mode – in this mode, backup occurs without temporarily suspend or powering off the VM (also known as live backup). This mode has the highest chance of error during backup, since it occurs while VM is running. There is no downtime but it takes the longest backup time.

Suspend mode – in this mode, backup occurs after temporarily suspending the VM. After backup is finished, the VM resume regular operation. This mode has lower chance of errors during backup since a VM is suspended. The downtime is moderate and it takes shorter backup time.

Stop mode – in this mode, backup occurs after powering off VM. After backup is finished, the VM resumes powered on. This mode has no chance of error during the backup, since the VM is not running at all. This is also the fastest backup mode.

SNAPSHOTS

This snapshots for VM is different with the snapshot mode for the full backup. It freezes the state of the VM in a point in time, and fully dependent on the original VM. Therefore, we cannot move snapshots somewhere for safekeeping. A best practice of this backup is using for testing purpose or applying updates. So, if something wrong happens, the previous state of VM can be simply reverted.

A backup file can be restored through the Proxmox VE web GUI or through the following CLI tools.

Backup compression

We can commit a backup with different compression levels. The higher the level, the less space is used to store backup files, but it also consumes higher CPU resources to perform compression. There are three compression levels in the Proxmox backup.

None – No compression, this take the least CPU during backup task.

LZO – This is default compression level in Proxmox. It provides balance between speed and compression. It also has the fastest decompression ratio, therefore making the restoration of a VM much faster.

GZIP – This provides the higher compression ratio but takes longer time to backup. Because of an increased compression ratio, it consumes a lot more CPU and memory resources.

VIRTUAL MACHINE REPLICATION

Proxmox has a very useful feature that is virtual machine replication. With this option, VMs can be real-time replicated to a different node in the cloud cluster. When the primary node goes down for any reason, the second node with a replica of the VMs can be brought online, therefore, minimizing downtime automatically. Furthermore, it is possible to schedule how regularly the VM will be replicated. The replication will start automatically without any user interaction. For example, the scheduled task can be set to run replicate a VM every 5 minutes.

3.4.4 What are Security measures are necessary for the Cloud environment?

Cloud features itself have been the solution to some of the traditional security, such as system downtime, backup, distributed storage or DDoS. Some solutions in the cloud are proposed to ensure safety as follows.

ACCESS CONTROL AND MANAGEMENT

Establishing a mechanism to control access is essential for information security to prevent unauthorized access. For example, assigning rights to users to use data and services. A note for this mechanism is to cover all processes of a user from the time of initial registration until the de-registration (user is no longer accessing the system and services). According to the Information Technology Infrastructure Library (ITIL) and ISO 27001/27002 standards for security, a Security management system must ensure the following functions [15]: Control access to information, Manage user access rights, Encourage good access practices, Control access to network services., Control access to operating systems, and Control access to applications and systems.

MEASUREMENT

One of the important points of cloud security is to find out the problems and security vulnerabilities that exist, then deploy appropriate measures to cope. In general, cloud systems are built on a set of multiple storage engines with high availability supported. That can be used to back up virtual servers, if something goes wrong. In order to gain flexibility, scalability, and performance, cloud providers face problems in analysing and calculating to reasonably allocate resources to various computational tasks.

Partitioning – An example to improve the computing performance of cloud-based applications is to split data into multiple partitions to perform calculations on multiple nodes. This way will increase the performance of queries and transactions. Therefore, the results are calculated and returned very quickly.

Migration – Flexibility is one of the main requirements of the cloud. For example, providing cloud services that need flexibility in resource usage. Resource must be reserved to the most essential and important activities. This makes the overload of nodes in the cloud does not occur when there is a migration of the system. Especially in the large database system, it still ensures the operation of the system when migration happens.

In addition, disaster recovery solutions should be taken into account when unexpected events occur such as natural disasters, floods, fires,

DDOS

It is clear that a system with full Firewall and IDS / IPS systems can still be DDoS attacked. However, if network infrastructure is strong enough, it can still suffer large DDoS traffic. Cloud infrastructure ensures this since the whole infrastructure is connected by a lot of computers. This gives administrators enough time to resolve the issue to find a fix. For example, the IPS system will learn new attack rules or administrators conduct packet analysis and set up rules to drop violated packets.

As of now, Snort and Suricata are two standard open-source IDS/IPS available, in spite of the fact that there are many others. One of the essential advantages od Suricata is that it is multithreaded, while Snort is single-threaded. Suricata is deployed faster and gain ubiquity in short amount of time. However, the limitation of Suricata is that there are no GUI options for Suricata in Proxmox. All configurations have to be done via the CLI. In addition, Suricata cannot be utilized to protect Proxmox nodes, only VMs. This restriction maybe due to the fact that IDS/IPS regularly consume a large amount of CPU resources. Whereas for a dedicated firewall appliance, this may or may not be an issue.

3.4.5 Design Principles

The design principles shown in the next subchapter are the concluding principles for every research question, that will be used in the design phase (functional and technical Design). The final choice of solution for every question will then also be determined in this phase.

1. What different kind of cloud solution are there and what is the best practice solution for the TinoHost cloud environment?

TinoHost environment will be an IAAS cloud. The requirement is to use an open-source platform to deploy the cloud, then in this case KVM (Kernel-based Virtual Machine) will be a suitable visualization technology.

The environment should be performance-intensive, in this case, Type 1 hypervisors are more light-weight, have better performance and introduce less latency.

If TinoHost have a research and develop team, then OpenStack is a solution. For our small deployment, then Proxmox will be a suitable platform, because Proxmox VE has built-in tools including hypervisor and User controller (Platfrom in a Flash). With these tools, you just need to download, flash it to the server and use it (GUI). Proxmox tightly integrates with KVM and LXC container.

2. What different kind of storage solution are there and what is the best practice solution for the TinoHost cloud environment?

With regards to the other design principles mentioning scalability and high availability, then distributed shared storage is a solution, because Proxmox require distributed shared storage to perform live migration and HA features. Ceph is a non-commercial option and it is tightly integrated in Proxmox GUI.

If we look for an alternative, then GlusterFS is a solution, because its performance is as same as Ceph but less popular and supported.

3. What are the different ways of doing proper backup and restore?

If we want a proper VMs back-up, then an automatic full back-up executed by scheduling in Proxmox GUI is a solution.

If we want freeze the state of the VM in a point of time, then a snapshot of the VM is a solution.

If we want to store a copy of the VM locally, then VM replication is a solution.

4. What are security measures are necessary for the cloud environment?
In order to protect the cluster, host nodes, and VMs, then creating firewall rule in the iptables is a solution. Because the Proxmox VE firewall is a security feature that allows easy and effective protection of a virtual environment for both internal and external network traffic.

In order to suffer large DDoS traffic, then using a dedicated firewall appliance, such as pfSense, Untangle, or any other open source firewall will be a solution.

3.5 Technical Design

This part of the technical design will describe the design of the complete infrastructure.

3.5.1 Technical requirements/ Design principles

The technical requirements can be found in the primarily phases/analyses of this design research document. The design principles can be found at the end of the literature research document, they will also be referred to in this technical design to explain the design choices.

3.5.2 Proxmox Deployment

For the prototype, because of hardware constraints VMware Workstation 15 will be used for deployment. First, I will test the features with a minimum configuration required, a three- node Proxmox cluster. However, in real practice a five-node cluster is highly recommended to archive the high performance as well as better fault tolerance ability (two nodes can fail).

Fault tolerance is how many members of the cluster can be down without causing the whole cluster to fail. As you can see in the table below the safest option is to have a cluster of 5-6 or more members. Then there will be enough instances that can take over as primary should the primary instance fail.

Number of nodes	Majority required to elect a new primary	Fault tolerance
3	2	1
4	3	1
5	3	2
6	4	2

Table 12: Proxmox fault tolerance

3.5.3 Ceph Deployment

Before deployment, it is important to take a look at some key components that make up a Ceph cluster and to have a proper understanding of what they are. There are five Ceph key components: RADOS, MONs (monitor), OSDs, Ceph Manager, and RGW. And three storage services: Block storage, Object storage, and File storage

RAM

Calculating the total RAM capacity of servers is vital to plan carefully. Initially, 64 GB of RAM is what we need. Any future upgrade would require to replace larger DIMMs. It requires at least 2 GB for each Ceph daemon (MON, RGW, MDS), and at least 2GB for each OSD.

⇨ Using at least 64GB RAM each server.

STORAGE DRIVES

Less-capacious drives are often cheaper from a price /GB (or price/TB) than more-capacious drives. Howevers, every drive needs a bay to live in therfore more-capacious drives can save more bay. Moreovers, less-capacious drives also present advantages against more-capaciuos drives. Like for example, a cluster built with 100 4TB drives may offer less aggregate speed (IOPS) than a cluster of the same capacity constructed from 400 1TB drives.

The HDD drives with the largest capacities often use SMR technology to achieve storage densities, which is present significant write pernalty. Unforatunately, this is not suitable for Ceph deployment, especially in latancy sensitive block storage implementation. Using caching may mitigate this drawback, but they are not the dives we are going to for.

⇨ For all the reason aboves, the choice will be SSD drives with the drive capacity either 512 GB or maximum 1TB to achieve more aggregate speed (IOPS).

Storage drive type

Since the SSD costs continue to fall, and they are increasing price competitive comparing to HDD drives (rotational drives). Considering to TCO, the rotational drives consume more power and also require more cooling. Thus, it narrows the financial gap.

⇨ Nowadays, enterprise SSD is more reliable than rotational drives. And our Ceph cluster role is to supply block storage to thousands of VMs running a variety of applications. But rotation drive cannot be satisfied with the performance limitation and it has a chance to running out of IOPS. Therefore, SSD drive is the choice for drive type

Storage drive durability and speed

Drive Write Per Day (DWPD) is a parameter that is not introduced much when we choose SSD quickly but this is the most important parameter and it indicates how long the SSD's durability is. However, very few buyers pay attention to what DWPD is. For example, a 1TB drive has a five-year warranty, 3 DWPD. That mean, if data write each day is 3TB, the drive can be used in 5 years. The more DWPD the more durability of the drive could be.

⇨ To decrease MON failures and improve longevity, I am strongly recommended to provision MON SSDs rate at least 3 DWPD or even better 10 DWPD.

TRADITIONAL INFRASTRUCTURE OR CONVERGED INFRASTRUCTURE

Traditional cloud and server farm designs typically have compute, storage, and network components are physically distinct and managed separately from each other. Whereas, converged infrastructure provisioning all services onto the same servers.

Pros: The advantages of this approach are saving in management cost and maintenance, in Data centre operational costs, as well as hardware up front capital costs.

Cons: The complicated of maintenance and the intertwining of infrastructure components. Conflict between competing services is also a concern that go up with scale and RAM utilization.

⇨ Converged infrastructure will be implemented in our design. For exampe, provisioning Proxmox VE and Ceph services onto the same server.

POOL DECISIONS

Proxmox cluster provides one or more pools to store data, for example, one for block storage and another for object storage. These pools have different design based on different use cases.

Replication

Keeping redundant copies of data is fundamental to Ceph. Ceph provides two redundancy strategies for ensuring data durability and availability: replication and erasure coding.

Replication pools in Ceph is mature and stable. It is familiar to who has worked with drive mirroring RAID 1 or RAID 10 (multiple copies of data are kept on multiple devices). The loss of replication is that the usable storage capacity of the cluster is a fraction of the raw capacity. Typically, replication defaults to making three copies of the data, which takes as much as three times storage than the original data before replication.

Data can be backed up by the most advanced mechanism available today, Erasure Coding. It is an alternative to replication pool to save storage space. Erasure Coding is a data protection method, in which data is divided into segments, extended and encoded with redundant pieces of data and stored on different locations or storage media.

OSD DECISIONS

In this section we will find out decisions that affect how OSDs is deployed within the Ceph clusters.

Backend: FileStore or BlueStore?

The Ceph OSD daemon has an important module called ObjectStore, which is responsible for object storage and object management. There are two ways that OSDs can manage the data they store. Starting from Luminious 12.2.z release (2017), the new default backend is BlueStore. Prior to Luminious, the default was FileStore.

In FileStore, objects are saved within a separate file. And by using FileStore, Ceph requires external journal to ensure consistency between data copies, all writes are seen as transaction units. Firstly, transactions are recorded in the journal. After writing to the journal, the FileStore daemon will write to the disk for storage.

However, using journal causes double write status (decrease ½ throughput of the hard drive), when journal and OSD share the same hard drive. If we set journal and OSD on 2 drives, when the hard drive containing the journal has problem, it will lead to all the OSD depending on that journal being lost. Therefore, from the Luminious release, Ceph had used a new storage backend, BlueStore, as the default. With this new storage mechanism, the data is stored directly to the raw drive, not through the file system layer, which eliminates journal from the design.

BlueStore was born to avoid the limitations of FileStore. With FileStore, the object must be written two times, once in the journal and once in the disk. BlueStore writes objects directly to disk and manages metada with RocksDB. Because RocksDB requires the use of a file systems, BlueStore uses file systems with the name BlueFS. There is a fact that, Ceph developer ensure that it’s at least twice faster than traditional FileStore and with less latency. In conclusion, BlueStore would have huge advantage against FileStore in terms of performance, disk I/O utilization.

⇨ BlueStore

OSD device strategy

It is strongly advice in production to deploy only one OSD per physical drive either with FileStore or BlueStore backend. Since Ceph is designed with this strategy in mind, neglecting it may cost issues on operational, device wear.

⇨ One OSD on one drive. It is important to plan for the cluster to not exceed 70% utilization of the raw OSD space.

OPERATING SYSTEM DECISIONS

When designing the operating system deployment for Ceph cluster, there are a number of elements to consider. Like the supported packages to be installed can be adjusted later. And specially the size and layout of boot drive partitions and filesystems. In general, it is best to carefully plan in advance when bootstrapping the system.

Kernel and operating system

Ceph is designed to be deployed on the modern Linux, determined user can build Ceph from source code, but most implementers deploy pre-built packages. Ceph is available .deb packages for Debian as well as Ubuntu, and as .rpm packages for RHEL and CentOS. SUSE also offers packages as well. It is practical to build on a newer kernel to enjoy the latest drives, bug fixes, and performance improvements.

⇨ Debian 10.0

Ceph packages

Most of Ceph admin downloads pre-built packages, which is from the central repository, for effortless installs and convenience. This approach ensures that the lasted released packages are always available. However, the problem of this approach is that it is potential if using HTTP instead of HTTPS, or dependency on potential slow and insecure internet connection. Besides that, there are some times the servers are offline for various reason and they are not allowed to connect to the external network because of local security policy.

Because of these reasons, it is better to maintain a local repository of Ceph packages, mirrored from the official website.

Use a repository management tool such as Create repo, Pulp, or Aptly. These tools allow us to mirror remote repositories, manage local package repo, take snapshot and pull new versions of packages along with dependencies.

NETWORKING DECISIONS

Ceph can work with either IPv4 OR IPv6 address, but not both at the same time. It is common for Ceph replications network to be provisioned with non-routable IPs such as RFC 1918, which is network internally for systems.

Ceph will use a public network for MONs and clients, and private network for OSDs replication traffic. The deployment of private network enjoys 40 Mbit links for each network, or it could be more cost effective with bonded 10 Mbit links. While bonding 25 Mbit show the promise for both cost and flexibility.

The public and private networks could be implemented with separate network switch or may use VLAN or other partition strategies. In the production model, it will be provision dual switches and cabling paths so that each network can use both capacity and fault tolerance. Each Ceph OSD node provision with dual port NICs, private and public bonds, so that even total failure of one NIC does not affected the OSDs on the server. With this bonding strategy, the bandwidth will be increased by configuring load balancing between two NICs and disruption is minimized if one link in a pair goes down.

Since we only deploy Ceph Block storage services, there is no need for network load balancer. When deploying the RGW object service a network load balancer is common.

According to best practice, it is recommended to have three physical separate networks in our design. Firstly, Public Network for communication between VMs and for connection to the Internet. Secondly, Proxmox Network (1Gbit\s, latency < 2ms) this network doesn't require wide bandwidth, so one gigabit network is sufficient. Lastly, Ceph Network (10Gbit\s) The third network need to have dedicated network for Ceph cluster and this network have to be at least 10Gbit. (New Quanta LB6M 10GbE 24 port Switch). The network is built with a redundant ability or active backup bindings.

3.5.4 TinoHost Cloud deployment model

As you can see in the design principles (more specifically design principle #1) we are going to design a Proxmox cloud environment since it gives us the most flexibility and security, especially for smaller teams.

In this part of the document we are going to describe the required infrastructure in detail.

VIRTUAL MACHINES

In this prototype, virtual machines are going to be used for testing purposes. But in the real practice, TinoHost is going to using standard bare-metal installation instead of virtual machines.

We are going to need separate VMs for the following:

- Proxmox cluster (Proxmox 6.0.1)
- Storage cluster (Ceph Nautilus V14.2.8)
- Backup cluster

NETWORKING

Here I will describe the network of the infrastructure for TinoHost, a multi-tenant cloud service provider. A diagram will also be provided. a virtualized firewall and virtual bridges are used to separate traffic between each client network. The virtual firewall has seven virtual network interfaces to connect client networks within a virtual environment and to provide WAN connectivity. Internet bandwidth is controlled through the virtual firewall for each vNIC. The virtual firewall is connected to WAN through the main virtual bridge, vmbr0. The Proxmox cluster has virtual bridges:

Subnet	Network description
vmbr0	Main virtual bridge to provide WAN connection to virtual firewall
vmbr1	Connects main storage cluster
vmbr5	Connects backup storage cluster
vmbr10	Bridge for company ABC subnet 10.10.10.0
vmbr20	Bridge for company XYZ subnet 10.20.20.0
vmbr30	Bridge for LXC containers for web hosting instances
Vmbr40	Bridge for a small business's virtual cluster/ Virtual private server (VPS)

Table 13: Network table

Each bridge connects the client company's virtual machines together and creates fully isolated internal networks from others.

DIAGRAM

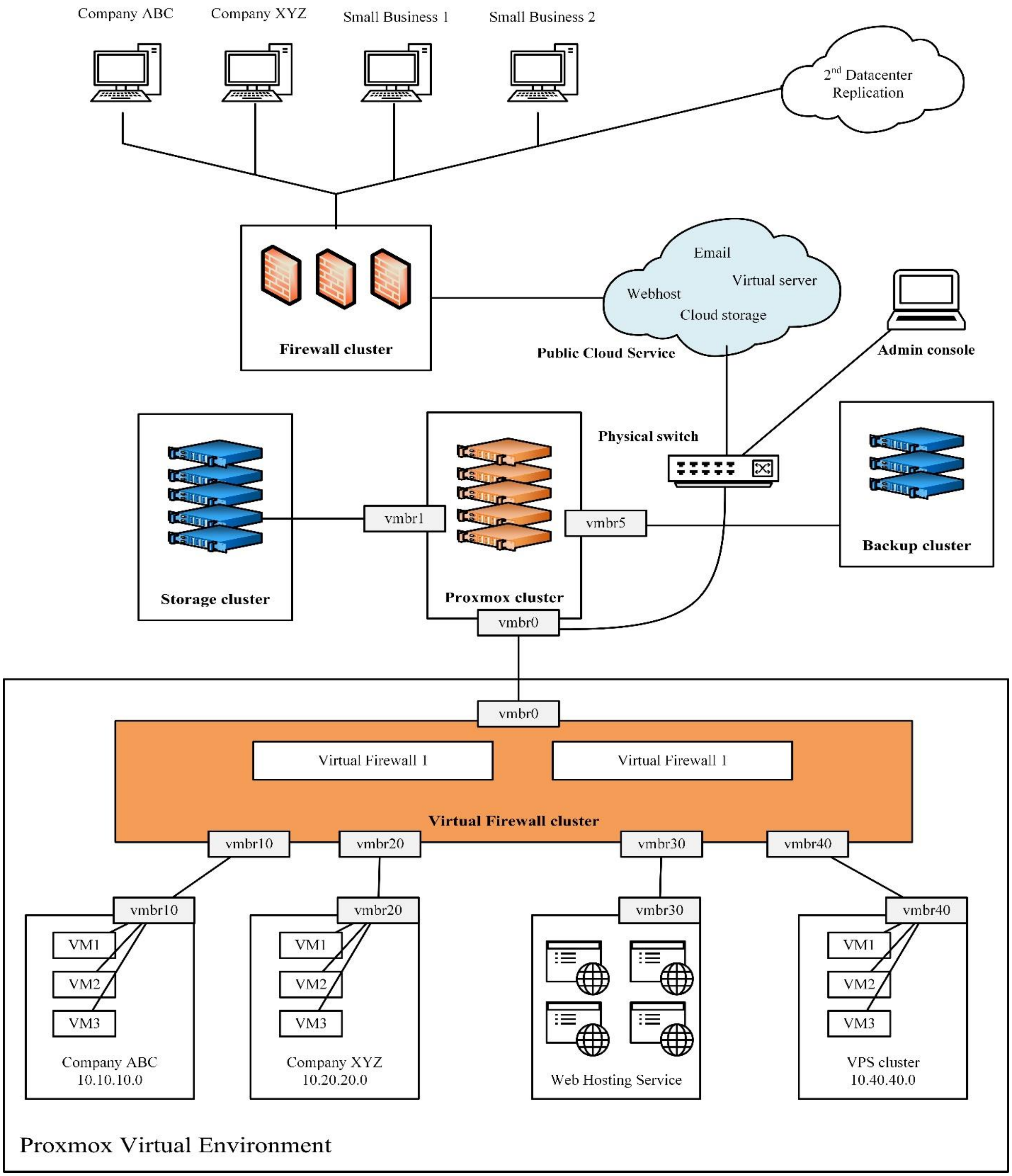


Figure 12: Network diagram of the complete TinoHost infrastructure

Shown above is the network diagram of the complete infrastructure that is recommended for the TinoHost environment. Evidently, all the servers are connected to their own corresponding router in their data centre.

3.5.5 Production-level set up

A production-level set up requires much-advanced planning and preparation than a testing stage. Once the setup is finished and the cluster has been brought online, it cannot be taken offline completely. Since the users who are using the cloud depend on it. In this section, I am going to illustrates a few of the key components or characteristics of the TinoHost production-level environment with multilayer redundancy, high performance, and stability.

STABLE AND SCALABLE HARDWARE

Stable hardware means minimum downtime. With quality hardware, it mitigates the risk of having randomized hardware failure in a cluster, causing unnecessary downtime. Therefore, choosing a good reputation brand of hardware is very important. Intel’s server components are well-known for stability and support, AMD hardware also a good choice, but factually AMD-based hardware has more stability issues. Besides stability, energy cost and heat generation are others two criteria for TinoHost decision. Intel CPUs use less energy and run much cooler then AMD competitor. AMD CPUs sue much higher wattage per CPU, which cause of high heat generation. For those reasons, choosing Intel CPU is a better choice comparing to AMD, since increased heat generation means an increased requirement for cooling, and therefore, increased utility bills. Although Intel products are more expensive, but the superb stability outweighs the higher cost per hardware.

Another deciding factor for hardware is scalable and availability. On the one hand, we will use hardware components which are easily find and always available when they need to be replaced. On the other hand, a good practice is using identical hardware for server nodes in the cluster based on their task. This makes hardware management simpler additionally permits in-hand stock build-up to rapidly replace a node when necessary.

REDUNDANCY

We can ensure that failure of a single component does not cause an entire service to fail by utilizing redundancy. There have to be redundancy in different levels of components.

NODE LEVEL

Node-level redundancy is limited to the node itself. It includes redundant power supply, network cards, RAID, and so on. With redundant power supply, the node can be connected to two different power sources, thus ensuring continuous operation during a power failure. Always use mirrored SSD drives as the operating system drive. During Proxmox installation we have an option to create a ZFS mirror on two physical drives. Or by using a physical RAID controller, we could also protect data and to avoid boot problems from the secondary mirror boot drive, if the primary boot drive fails. These two options will ensure that the operating system itself will run uninterrupted, even if a drive fails entirely.

An entire data centre can be failed if, for example, a natural disaster strikes a particular geographic region. In this case, a well-designed multiple data centre replication topology can prevent an entire service from becoming unavailable. At this time, we are not going to do this because of the cost limitation.

UTILITY LEVEL

In order to the cluster keep running during power loss, it is necessary to provide some sort of backup power. At the moment, TinoHost is renting a rack space in Viettel IDC which is the largest Tier-3 data centre in Vietnam. The data centre offers 2(N+1) redundancy that means it has double the amount of power equipment needed, plus an additional UPS module on each side. 2(N+1) data centres offer the highest guaranteed uptime, therefore, we do not have to care much about the power loss problem.

NETWORK LEVEL

By using multiple network interfaces, switches and network paths, we ensure that network connectivity will not be interrupted during a switch or cable failure. Redundant cables are need to protect again cable failure, each network interface card is connected to a different cable, and the cables themselves are connected by a network device such as a hub or a bridge. Layer 3 switches[1], such as stackable switches, could be used to create the right redundant network paths. They are ideal for VLANs only, as they do not have a WAN interface. But within VLANs, it offers multiple options to manage bandwidth efficiently. This is why layer 3 switches are powerful and scalable technology for building high-performance Ethernets.

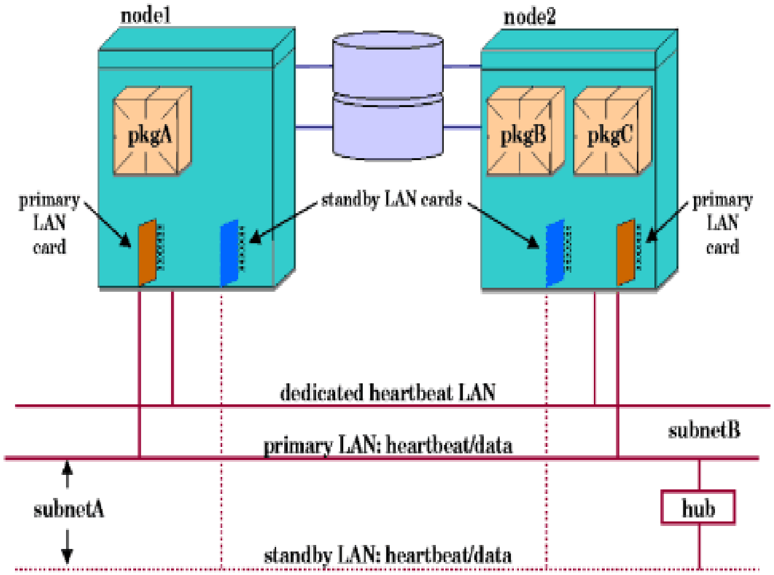


Figure 13: Redundant LANs [1]

STORAGE LEVEL

In the production level, Ceph enterprise-grade storage system is going to be used. Because Ceph has redundancy built into the firmware. We still need to ensure storage nodes in the cluster have different-level redundancy in place.

[1] Layer 3 switches act as both switches and routers.

CURRENT LOAD AND FUTURE GROWTH

Taking into account failover load is crucial when provisioning node memory configuration in our design. In our five-node cluster design, we allow two nodes of a five-node cluster to fail. Therefore, I would want each node to use only 60 percent of the available memory. For example, five nodes in Proxmox cluster have 320 GB RAM (64 GB RAM each node), and 192 GB will be the maximum memory is consumed at all times by all the virtual machines.

Future growth is also an important factor in design a cluster. TinoHost cluster must be able to scale easily with the company growth and adapt to increased workloads requirement. Both Proxmox and Ceph clusters have ability to scale at any time and any to size by simply add new hardware nodes. Thus, increasing the resources required by the VMs.

TRACKING HARDWARE INVENTORY

For tracking hardware inventory, then creating a spreadsheet would be enough to keep track of all key information about hardware being used in the network. In the near future, if the size of cluster is expanded enormously, we will implement a proper tracking system. By doing that way, time can be saved a lot when any information needs to be retrieved.

3.6 Design Evaluation

This evaluation document consists of two parts: the design evaluation and the final evaluation itself. These evaluations will answer the question whether the intervention solve the problem. In this design evaluation, it will be looked at if the design fits in sufficiently with the conclusions in the preliminary research

Methodology

The prototype of the environment is going to be tested by the test scenarios. Test scenarios are actual tests of the more advanced solutions meant for the production environment, such as high availability and scalability solutions, were executed.

3.6.1. Result

In the design evaluation phase, it is evaluated if the intervention solved the problem, by reviewing if the design matches the conclusions of the preliminary research.

TEST SCENARIOS

Test ID	1
Scenarios	Check live migration with Ceph
Tested Component	VM will be migrate between node without data loss.
Test Description	A lightweight VM will be created on node 1. I will create a simple text file and use it to prove live migration works.
Desired result	When the VM is migrated to node 2. It works perfectly, the text file is there. Then I modified the text file and migrate the VM to node 3. As expected, the VM work perfectly and the modified text file still there.

Test ID	2
Scenarios	Check High availability / Automatic failover feature
Tested Component	For automatic failover I will use HA. HA ensures that VMs are restored on other nodes whenever the host node is down.
Test Description	I will kill the host node. When the node is down, it should take some time for HA to restore the VM and the container on another node.
Desired result	The VM has been restored on the node 3 while the container has been restored on node 2. The resources are limited, in the production environment it would be expected a much smoother transition. If the host node is brought back again, the VM and container will not migrate back unless I do it manually or another node goes down.

Test ID	3
Scenarios	Low amount of storage available on the cluster.
Tested Component	Storage scalability
Test Description	Add an extra node to the cluster
Desired result	Increase amount of available space, the cluster recognises the new node and starts utilizing it.

Test ID	4
Scenarios	Check if the Proxmox network connectivity works.
Tested Component	Network connectivity
Test Description	Try to ping each other and access to Proxmox GUI.
Desired result	Connectivity on network interface

Test ID	5
Scenarios	Check if the backup works.
Tested Component	Backup images
Test Description	Try to backup VMs in one node
Desired result	The VMs are backup and can be restore correctly.

3.6.2. Conclusion

In this chapter, I will explain our findings about the design based on the data from the test scenarios.

RESULT FROM THE TEST SCENARIOS

Test ID	“Happy Day” Result	Potential Failure Result
1	Live migration working correctly.	Proxmox takes too much time to migrate VMs and containers.
2	HA/ Automatic failover is working correctly.	When a node goes down, VMs and containers are not be move to other healthy nodes automatically.
3	When the extra node is added it is configured correctly, it is recognized by the master and it is utilized.	The extra node is added, but it is not configured correctly, the master does not recognize it and is not utilized.
4	Connectivity is properly configured and data is retrieved.	Connectivity is not properly configured and no data is received.
5	VMs and LXC containers are backing up and restored properly.	A back task takes a very long time to complete, or it crashes when multiple nodes are backing up to the same backup storage.

CONCLUSION

As can be seen in the possible outcomes above the system should handle everything correctly if the initial configuration of it is correct. Having to remember to include the human factor in every system since even if we have the best technologies available, if they were not enabled and/or configured correctly by the system administrators they will simply not work.

I am confident that the system is resistant to failure and that it can recover itself automatically when the need occurs. The system is intelligent and knows when actions need to be taken, again, if configured correctly.

3.7 Final Evaluation

This final evaluation will answer the question if the provided solutions are a definitive answer to the problem statement:

How do we choose the best fit open source Infrastructure as the Service (IaaS) solution for building and managing clouds?

First, all the sub questions will be evaluated. Finally, in the conclusion, an answer will be given to the main problem statement.

3.7.1 Hypervisor solution

What different kind of Hypervisor are there and what is the best practice solution for TinoHost?

As already stated in the technical design, Proxmox VE was chosen as the final solution for the hypervisor of the CC design. This This is because Proxmox is the best suitable choice when it comes to compatibility with the solution of the prototype.

3.7.2 Storage solution

What different kind of storage solutions are there and what is the best practice solution for TinoHost?

In combination with Proxmox VE, Ceph is the best practice storage option at the moment, because of its unified, distributed, cost-effective, and scalable nature is the potential solution to today's and the future's data storage needs.

3.7.3 High Availability

How will high availability with an uptime of 98% be realized?

Inbuild Proxmox HA feature will be in place to automatically moving or migrating VMs to a node as soon as server hardware failure occurs. HA does not provide zero downtime but it minimizing downtime as little as possible. In addition, the administrators do not have to manually move or migrate any VMs from a faulty node.

3.7.4 Security solution

What security measures are necessary for the design with regards to the networking?

When it comes to security, there are the following most important points:

Firewall: By creating firewall rules we can protect VMs, host nodes or the entire cluster. Although the Proxmox firewall (iptables) provides excellent protection, it is highly recommended to have a physical firewall (edge firewall[2]) for the entire network.

Logging: Logging allows us to see what is happening with the infrastructure. We can easily detect suspicious activities and trace them to their origin.

[2] Physical firewall or edge firewall sits at the main entry point to the internet

Access control: It is important for proper access control to be in place. Only the system administrators should be able to access the production servers and passwords should be stored in a safe way.

3.7.5 Backup solution

What are the different ways of doing a proper data backup and/or snapshot to saving the data?

A Proxmox cluster with redundancy node to backup is a recommend practice. Therefore, a dedicated NFS backup storage node should be in place for the backup images instead of local storage. By doing this way, the backup location is centralized and easier to restore VMs in event of a Proxmox node failure. If the backup stored locally in the Proxmox node, it will become inaccessible during the failures. The primary goal is to store a backup on a separate node instead of the computing node. In Proxmox, automatic backups could be achieved by creating a schedule for backup. Schedule can be created from the Backup option in the GUI. This backup solution should be used to create a good data disaster recovery plan.

4. CONCLUSION AND FUTURE WORK

How do we choose the best fit open source Infrastructure as the Service (IaaS) solution for building and managing clouds?

I believe that this design provided a suitable final solution for TinoHost. By choosing for the Proxmox VE cloud environment, the infrastructure of the CC is future-proof. Furthermore, not a lot of IT staff is necessary to maintain this infrastructure. Finally, Proxmox VE offers excellent scalability and high availability solutions that are fully automatized.

The work has been achieved

- Completed installation and operation of Proxmox and Ceph, with the two most important services being providing resources on Cloud compute and Cloud storage.
- Uploaded and tested basic Linux virtual machines and container like Ubuntu, Debian, CentOs.
- Tested HA, live migration, and backup features through GUI and command line.

The work has not been achieved

With computational resources: it is now possible to run fairly stable Linux virtual machine instances. However, virtual machines running Windows have not been successful. There are still many errors such as losing an instance or not being able to log into the instance.

With limited storage resources, only a preliminary test is performed, no deep understanding of the ability to expand storage, archive large files. In addition, due to hardware conditions, deployment time, it has not been tested performance.

Plan in the next step

The new version of Proxmox, released in early December 2019, promises a lot of improvements and more stability. The next plan is to switch to testing this version, combined with the expansion of physical infrastructure. With a larger model, the ability to test the performance of components in the cluster will be more detailed and evaluated more accurately.

APPENDIXIES

Any documentation written related to the internship but not directly to this specific report are being represented within this appendixes chapter.

Appendix A: Proxmox cluster installation

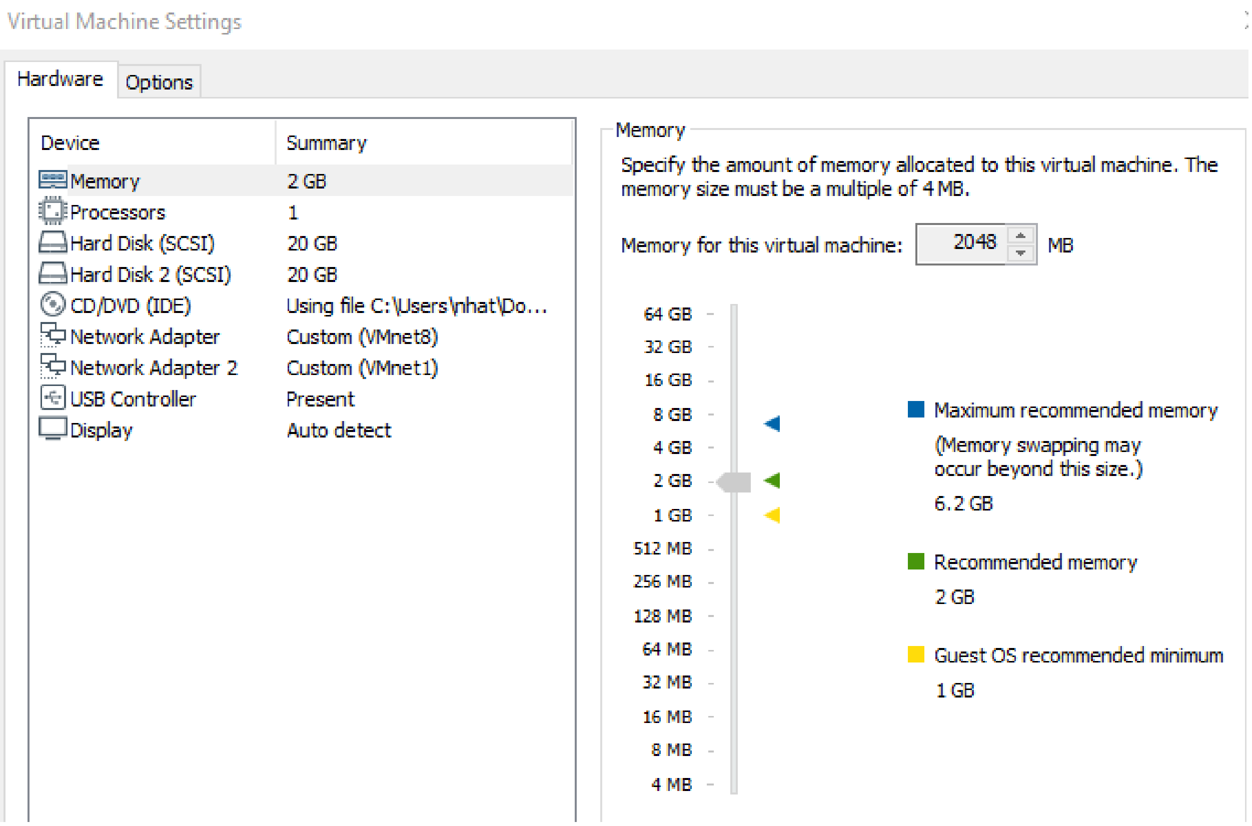
This appendix will be looking at the new features of Proxmox VE 6.0 including creating and joining cluster; Ceph installation, Live migrations and HA.

This demo had been done on my notebook which is installed VMware workstation 15. The notebook specifications are:

- CPU intel core i7 7700 HQ
- Ram 16 GB
- Storage 200 GB

For this demo, 3 nodes are needed for HA and live migrations feature. In this case I also created three-node Proxmox cluster.

Step 1: Creating nodes



Virtual Machine Settings

Device	Summary
Memory	2 GB
Processors	1
Hard Disk (SCSI)	20 GB
Hard Disk 2 (SCSI)	20 GB
CD/DVD (IDE)	Using file C:\Users\nhat\Do...
Network Adapter	Custom (VMnet8)
Network Adapter 2	Custom (VMnet1)
USB Controller	Present
Display	Auto detect

Memory

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

Memory for this virtual machine: 2048 MB

🟦 Maximum recommended memory
(Memory swapping may occur beyond this size.)
6.2 GB

🟩 Recommended memory
2 GB

🟨 Guest OS recommended minimum
1 GB

- 2 hard disks for raid 1

- 2 NIC (1 NAT (vmnet8) and 1 host only(vmnet1)). We need two networks for Cluster and Ceph redundancy.

After that install Proxmox via iso and choose advance option for Raid 1.

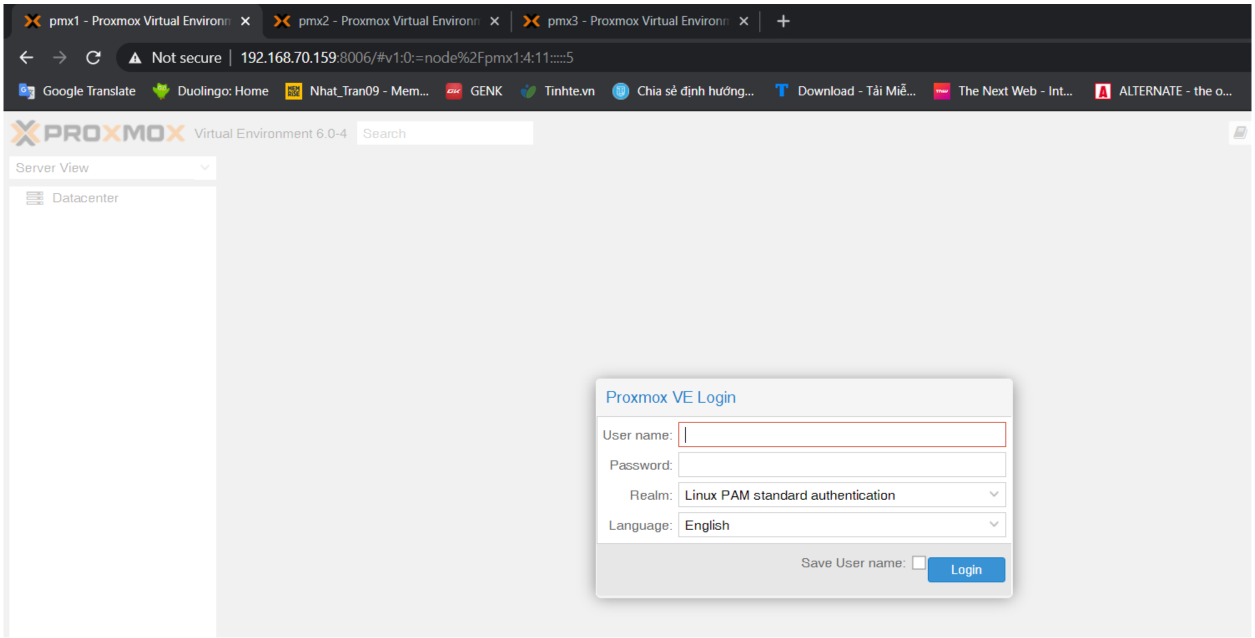
User name is “root” and password for the node will be “minhnhat”.

Clone the other two nodes from the first node:

To avoid having to setup everything again, I will clone the VM. This will also ensure that I have the same features on the 3 nodes. In order to clone the VM, we need to turn off the VM. Then clone it into node2 and node3.

Step 2: Preparing Proxmox for clustering and Ceph

Go to Proxmox GUI via the IP address and login with username and password above.



Not secure | 192.168.70.159:8006/#v1:0:=node%2Fpmx1:4:11:::::5

Proxmox VE Login

User name:

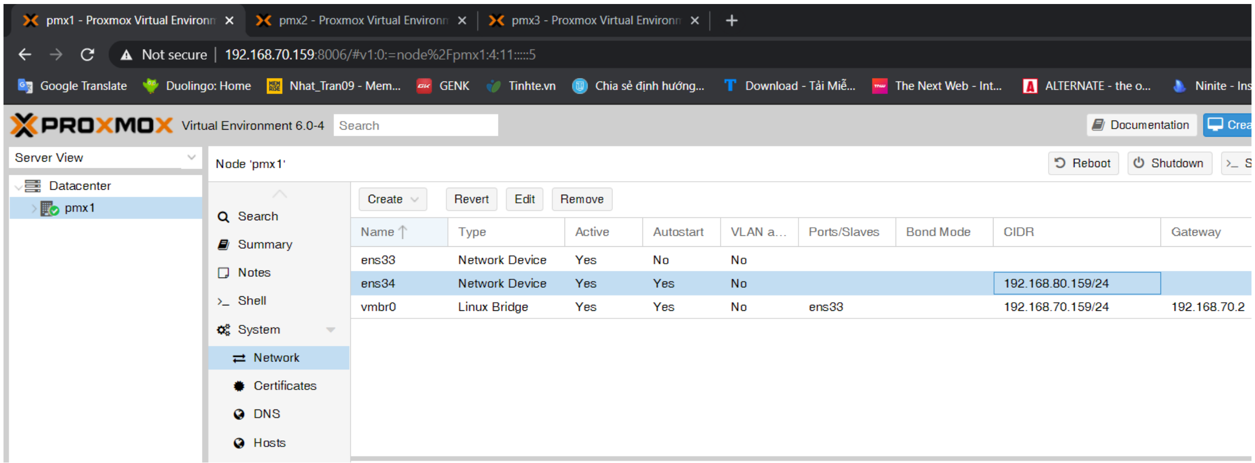
Password:

Realm: Linux PAM standard authentication

Language: English

Save User name: □

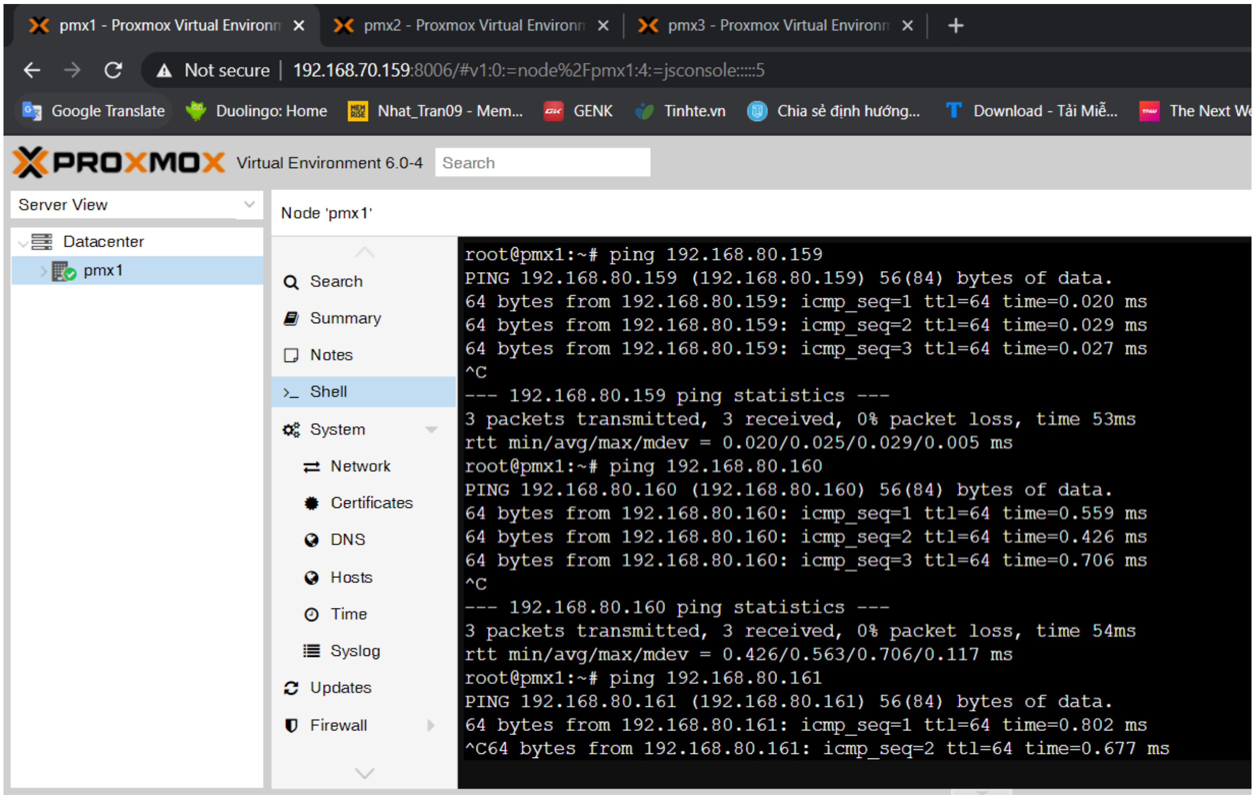
All the three nodes are ready and are accessible on the network. I will now setup the secondary network for all the nodes. It is only needed to setup the IPs for the network and reboot them.



pmx1 - Proxmox Virtual Environm
pmx2 - Proxmox Virtual Environm
pmx3 - Proxmox Virtual Environm
Not secure | 192.168.70.159:8006/#v1:0:=node%2Fpmx1:4:11::::5
PROXMOX Virtual Environment 6.0-4
Documentation
Server View
Datacenter
pmx1
Node 'pmx1'
Reboot
Shutdown
Search
Summary
Notes
Shell
System
Network
Certificates
DNS
Hosts
Create
Revert
Edit
Remove

Name	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway
ens33	Network Device	Yes	No	No				
ens34	Network Device	Yes	Yes	No			192.168.80.159/24	
vmbr0	Linux Bridge	Yes	Yes	No	ens33		192.168.70.159/24	192.168.70.2

And then check the connection between nodes by ping each other.

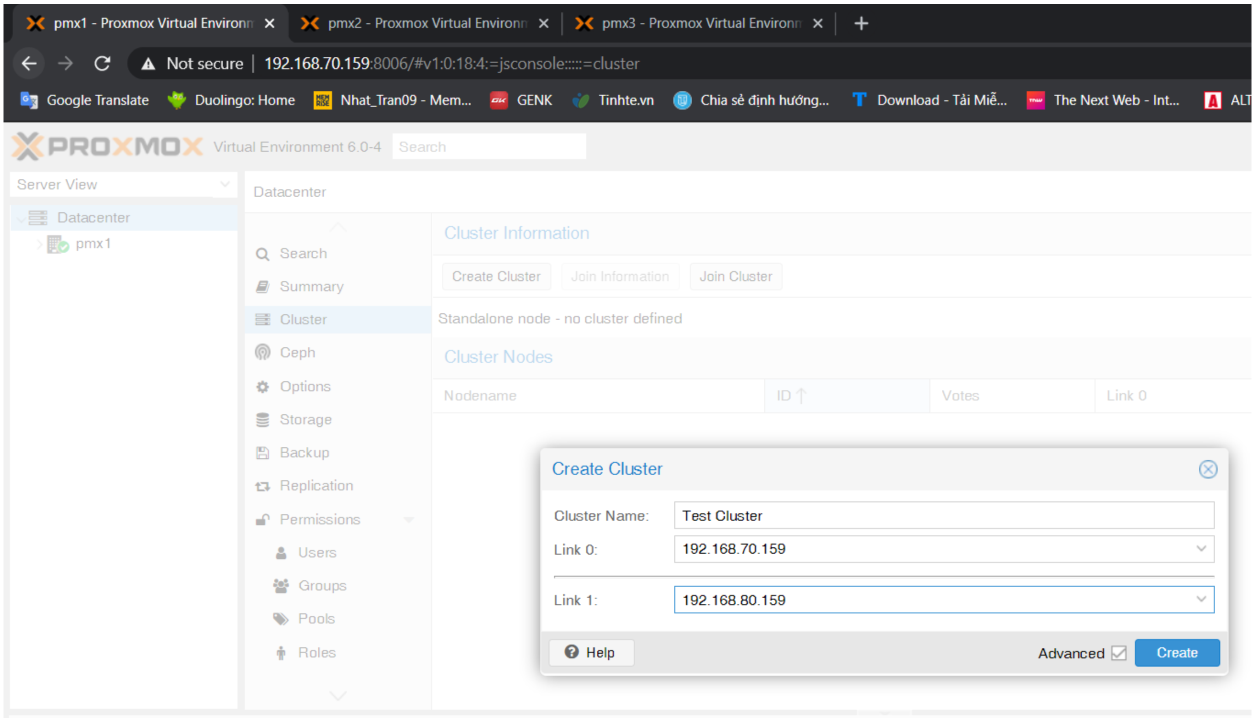


pmx1 - Proxmox Virtual Environm
pmx2 - Proxmox Virtual Environm
pmx3 - Proxmox Virtual Environm
Not secure | 192.168.70.159:8006/#v1:0:=node%2Fpmx1:4:=jsconsole::::5
PROXMOX Virtual Environment 6.0-4
Server View
Datacenter
pmx1
Node 'pmx1'
Search
Summary
Notes
Shell
System
Network
Certificates
DNS
Hosts
Time
Syslog
Updates
Firewall

```
root@pmx1:~# ping 192.168.80.159
PING 192.168.80.159 (192.168.80.159) 56(84) bytes of data.
64 bytes from 192.168.80.159: icmp_seq=1 ttl=64 time=0.020 ms
64 bytes from 192.168.80.159: icmp_seq=2 ttl=64 time=0.029 ms
64 bytes from 192.168.80.159: icmp_seq=3 ttl=64 time=0.027 ms
^C
--- 192.168.80.159 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 53ms
rtt min/avg/max/mdev = 0.020/0.025/0.029/0.005 ms
root@pmx1:~# ping 192.168.80.160
PING 192.168.80.160 (192.168.80.160) 56(84) bytes of data.
64 bytes from 192.168.80.160: icmp_seq=1 ttl=64 time=0.559 ms
64 bytes from 192.168.80.160: icmp_seq=2 ttl=64 time=0.426 ms
64 bytes from 192.168.80.160: icmp_seq=3 ttl=64 time=0.706 ms
^C
--- 192.168.80.160 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 54ms
rtt min/avg/max/mdev = 0.426/0.563/0.706/0.117 ms
root@pmx1:~# ping 192.168.80.161
PING 192.168.80.161 (192.168.80.161) 56(84) bytes of data.
64 bytes from 192.168.80.161: icmp_seq=1 ttl=64 time=0.802 ms
^C64 bytes from 192.168.80.161: icmp_seq=2 ttl=64 time=0.677 ms
```

Step 3: Creating and joining the cluster

Giving the cluster a name “TestCluster” then points it to the primary network. The secondary network is also important for redundancy.



Standalone node - no cluster defined

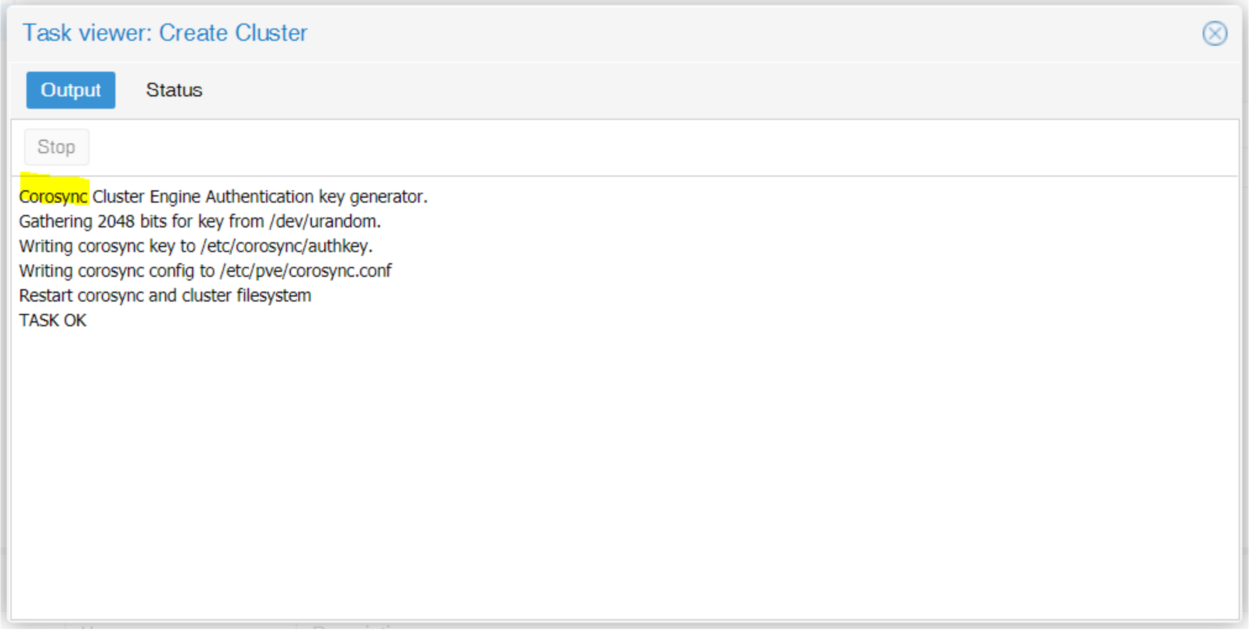
Create Cluster

Cluster Name: Test Cluster

Link 0: 192.168.70.159

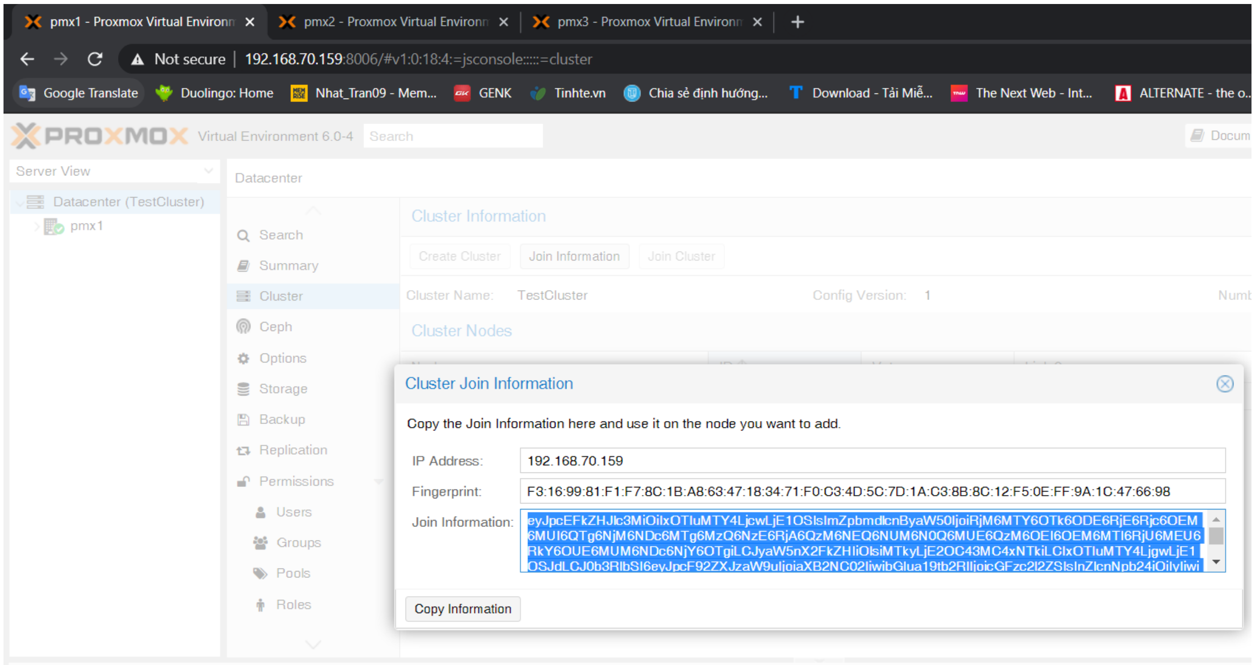
Link 1: 192.168.80.159

The cluster is created easily with very simple GUI process. Proxmox use Corosync for reliable group communication to create clusters.



```
Corosync Cluster Engine Authentication key generator.
Gathering 2048 bits for key from /dev/urandom.
Writing corosync key to /etc/corosync/authkey.
Writing corosync config to /etc/pve/corosync.conf
Restart corosync and cluster filesystem
TASK OK
```

In order to join other nodes to the cluster, I will copy the joining information which will be needed by the other nodes.



Cluster Name: TestCluster Config Version: 1

Cluster Join Information

Copy the Join Information here and use it on the node you want to add.

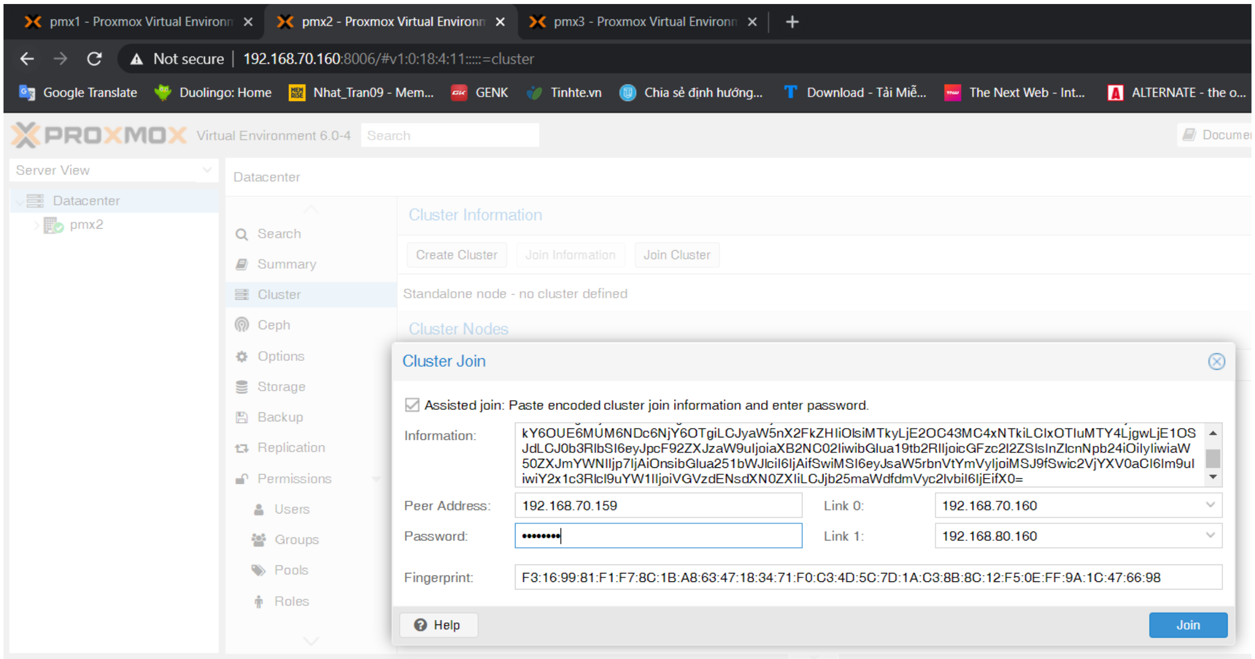
IP Address: 192.168.70.159

Fingerprint: F3:16:99:81:F1:F7:8C:1B:A8:63:47:18:34:71:F0:C3:4D:5C:7D:1A:C3:8B:8C:12:F5:0E:FF:9A:1C:47:66:98

Join Information:

eyJpcEFkZHJlc3MiOilxOTIuMTY4LjcwLjE1OSIsImZpbmdlcnByaW50IjoiRjM6MTY6OTk6ODE6RjE6Rjc6OEM
6MUI6QTg6NjM6NDc6MTg6MzQ6NzE6RjA6QzM6NEQ6NUM6N0Q6MUE6QzM6OEI6OEM6MTI6RjU6MEU6
RkY6OUE6MUM6NDc6NjY6OTgiLCJyaW5nX2FkZHIiOlsiMTkyLjE2OC43MC4xNTkiLCIxOTIuMTY4LjgwLjE1
OSJdLCJ0b3RlbSI6eyJpcF92ZXJzaW9uIjoiaXB2NC02IiwibGlua19tb2RlIjoicGFzc2l2ZSIsInZlcnNpb24iOilyIiwi

On the second node, click on the “Join Cluster” button then paste the info copied from the node 1. Then selecting both the primary and secondary networks for the cluster and provide the password for node 1 in order to join.



Standalone node - no cluster defined

Cluster Join

☑ Assisted join: Paste encoded cluster join information and enter password.

Information:

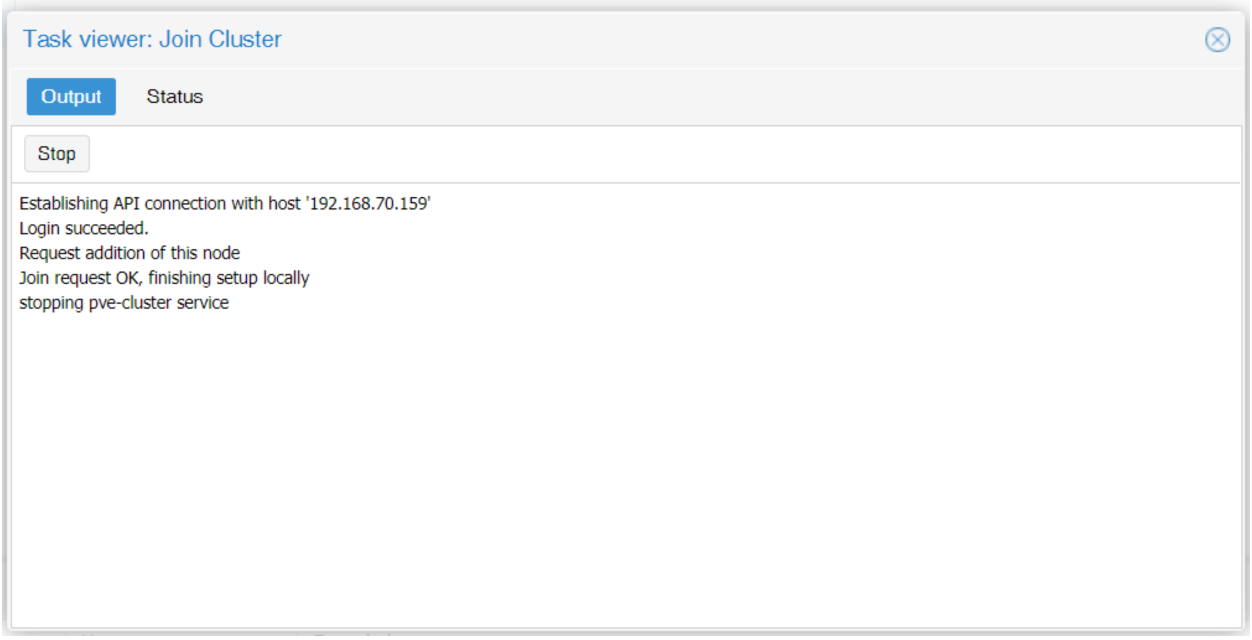
kY6OUE6MUM6NDc6NjY6OTgiLCJyaW5nX2FkZHIiOlsiMTkyLjE2OC43MC4xNTkiLCIxOTIuMTY4LjgwLjE1OS
JdLCJ0b3RlbSI6eyJpcF92ZXJzaW9uIjoiaXB2NC02IiwibGlua19tb2RlIjoicGFzc2l2ZSIsInZlcnNpb24iOilyIiwiaW
50ZXJmYWNlIjp7IjAiOnsibGluazE1bWJlcil6IjAifSwiMSI6eyJsaW5rbnVtYmVyIjoiMSJ9fSwic2VjYXV0aCI6Im9uI
iwiY2x1c3Rlcl9uYW1lIjoiVGVzdENsdXN0ZXIiLCJjb25maWdfdmVyc2lvbiI6IjEifX0=

Peer Address: 192.168.70.159 Link 0: 192.168.70.160

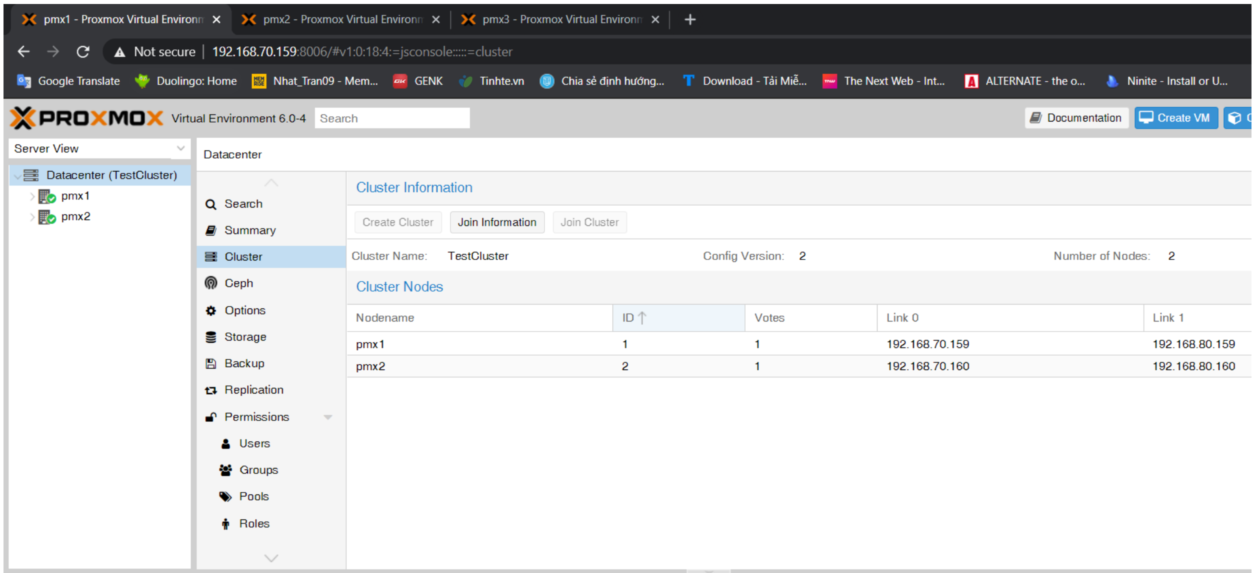
Password: •••••••• Link 1: 192.168.80.160

Fingerprint: F3:16:99:81:F1:F7:8C:1B:A8:63:47:18:34:71:F0:C3:4D:5C:7D:1A:C3:8B:8C:12:F5:0E:FF:9A:1C:47:66:98

Just like that, I am done with node 2. It is a very simple process and no more commands needed. The cluster is up and running.



```
Establishing API connection with host '192.168.70.159'
Login succeeded.
Request addition of this node
Join request OK, finishing setup locally
stopping pve-cluster service
```

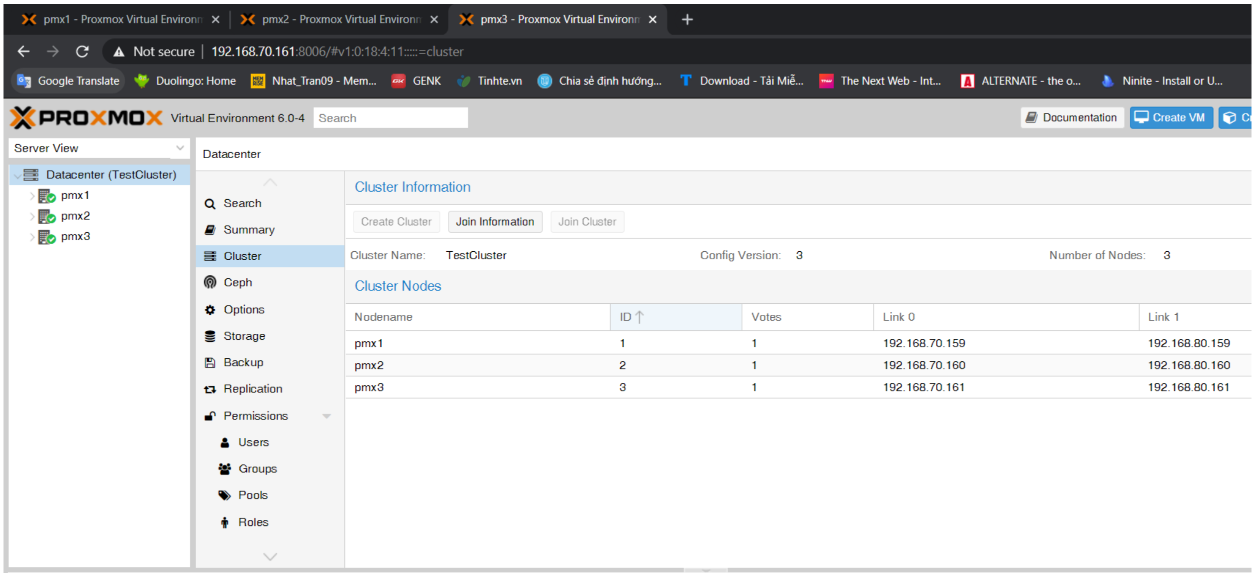


Cluster Name: TestCluster Config Version: 2 Number of Nodes: 2

Cluster Nodes

Nodename	ID ↑	Votes	Link 0	Link 1
pmx1	1	1	192.168.70.159	192.168.80.159
pmx2	2	1	192.168.70.160	192.168.80.160

The process repeated the same on node 3; copy the joining information and paste it in the node's joining window.



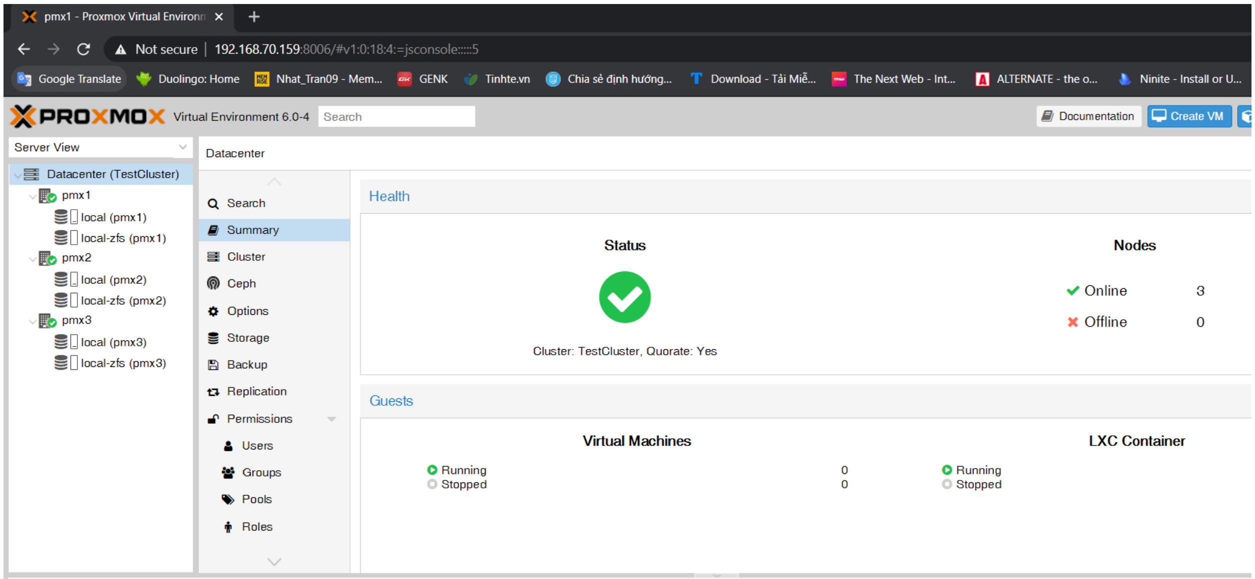
Cluster Name: TestCluster

Config Version: 3

Number of Nodes: 3

Nodename	ID ↑	Votes	Link 0	Link 1
pmx1	1	1	192.168.70.159	192.168.80.159
pmx2	2	1	192.168.70.160	192.168.80.160
pmx3	3	1	192.168.70.161	192.168.80.161

The three-node cluster setup is done. I do not need the 3 windows open; I can now access all 3 nodes from on window.



Status

Cluster: TestCluster, Quorate: Yes

Nodes

✓ Online 3

✗ Offline 0

Virtual Machines

○ Running 0

○ Stopped 0

LXC Container

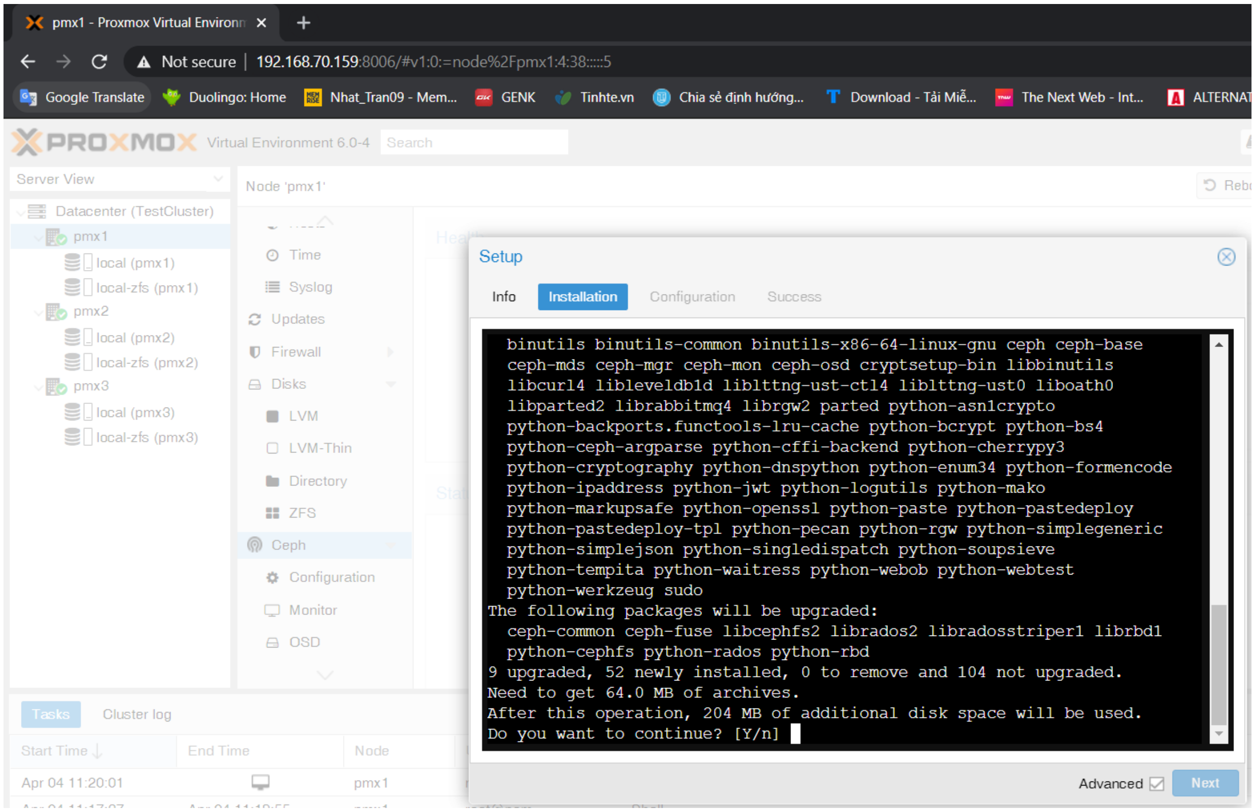
○ Running

○ Stopped

Next a Ceph distributed storage cluster will be created and then integrated with Proxmox cluster to ensure High Availability and live migration features.

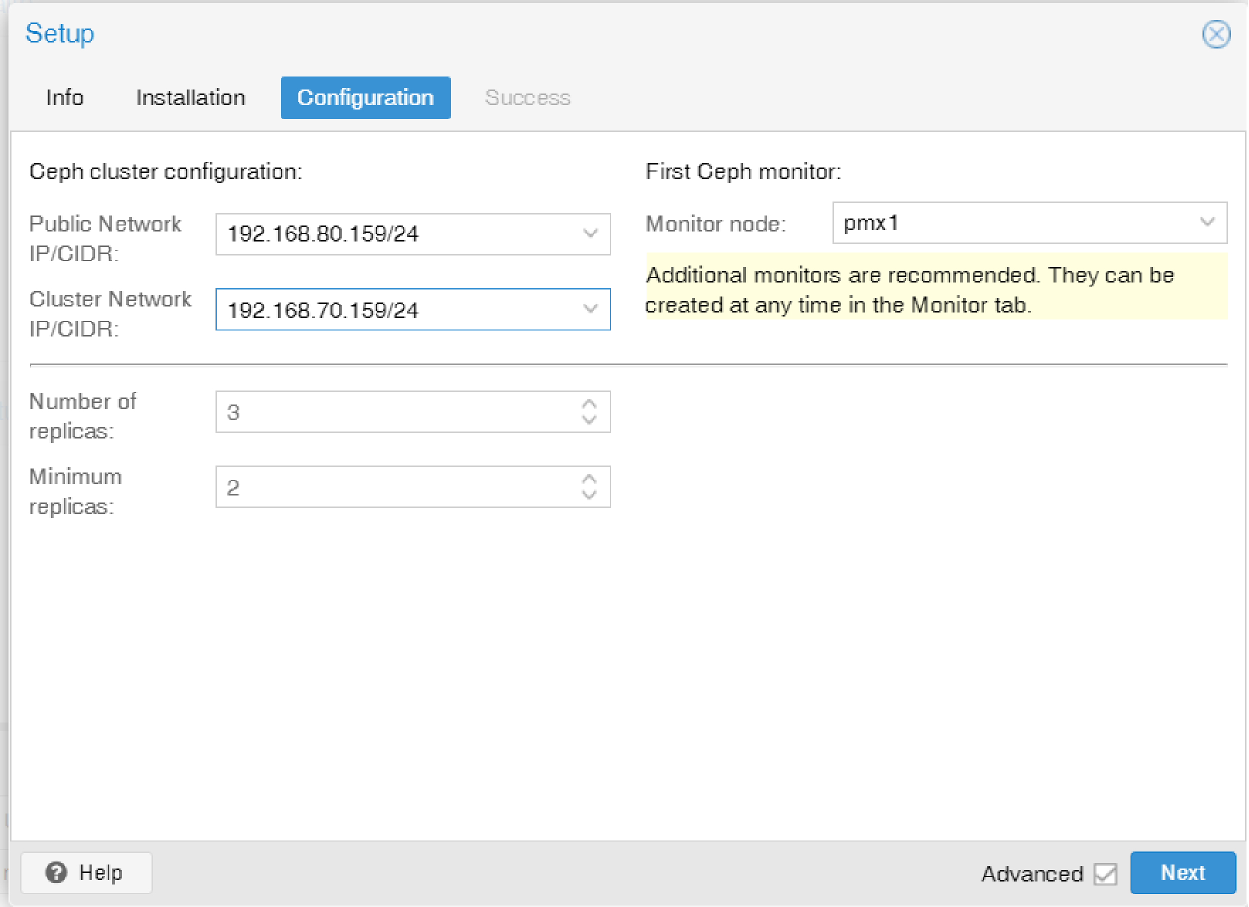
Step 4: Ceph creation

The process goes through a simple wizard to create the monitors. The wizard will download the necessary packages and then proceed with the installation.



```
  binutils binutils-common binutils-x86-64-linux-gnu ceph ceph-base
  ceph-mds ceph-mgr ceph-mon ceph-osd cryptsetup-bin libbinutils
  libcurl4 libleveldb1d liblttng-ust-ctl4 liblttng-ust0 liboath0
  libparted2 librabbitmq4 librgw2 parted python-asn1crypto
  python-backports.functools-lru-cache python-bcrypt python-bs4
  python-ceph-argparse python-cffi-backend python-cherrypy3
  python-cryptography python-dnspython python-enum34 python-formencode
  python-ipaddress python-jwt python-logutils python-mako
  python-markupsafe python-openssl python-paste python-pastedeploy
  python-pastedeploy-tpl python-pecan python-rgw python-simplegeneric
  python-simplejson python-singledispatch python-soupsieve
  python-tempita python-waitress python-webob python-webtest
  python-werkzeug sudo
The following packages will be upgraded:
  ceph-common ceph-fuse libcephfs2 librados2 libradosstriper1 librbd1
  python-cephfs python-rados python-rbd
9 upgraded, 52 newly installed, 0 to remove and 104 not upgraded.
Need to get 64.0 MB of archives.
After this operation, 204 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

I will then provide the public network and the cluster network. For the public network, I will use secondary network.



Setup

Ceph cluster configuration:

Public Network IP/CIDR: 192.168.80.159/24

Cluster Network IP/CIDR: 192.168.70.159/24

Number of replicas: 3

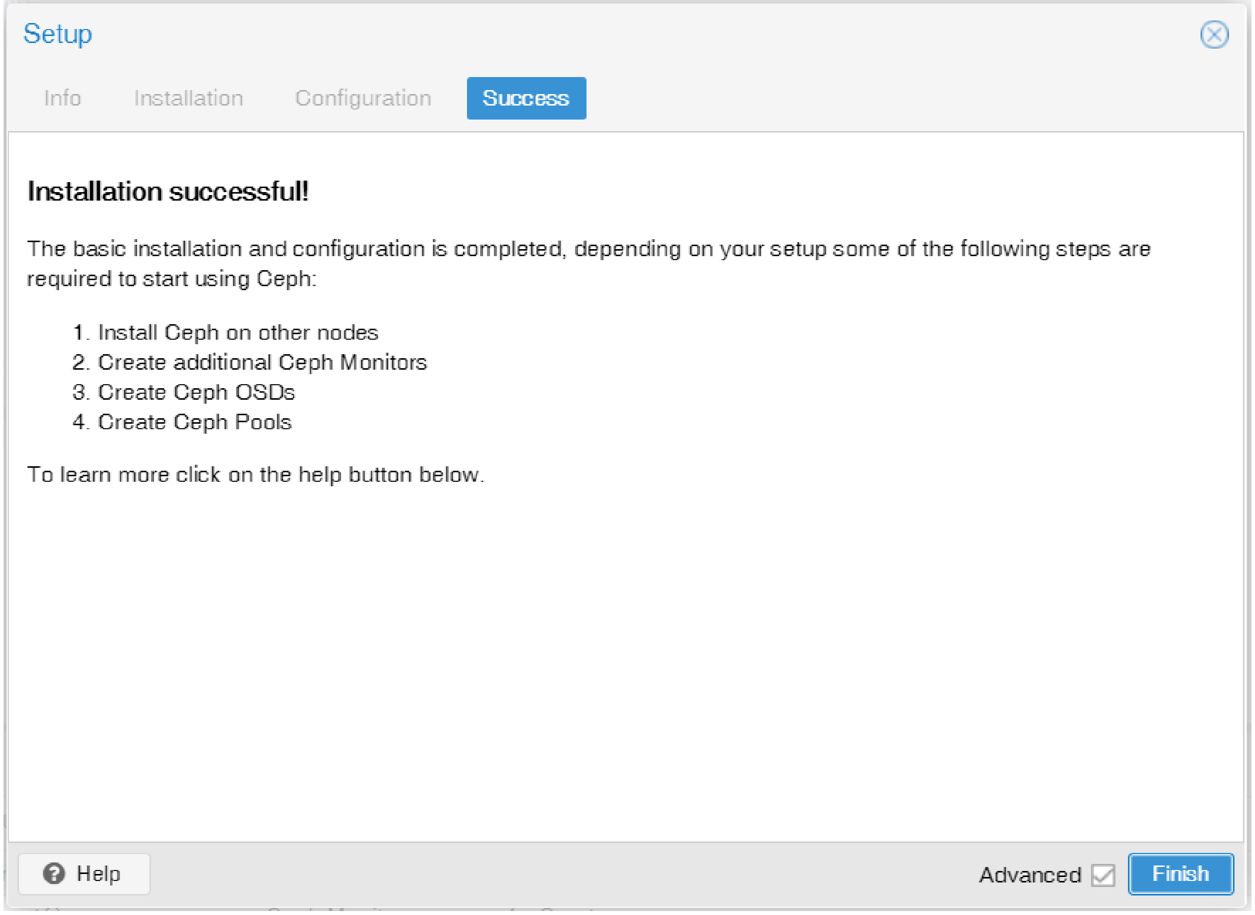
Minimum replicas: 2

First Ceph monitor:

Monitor node: pmx1

Additional monitors are recommended. They can be created at any time in the Monitor tab.

The first Ceph monitor is setup. I now need to do the same on the two other nodes to complete the setup.



Installation successful!

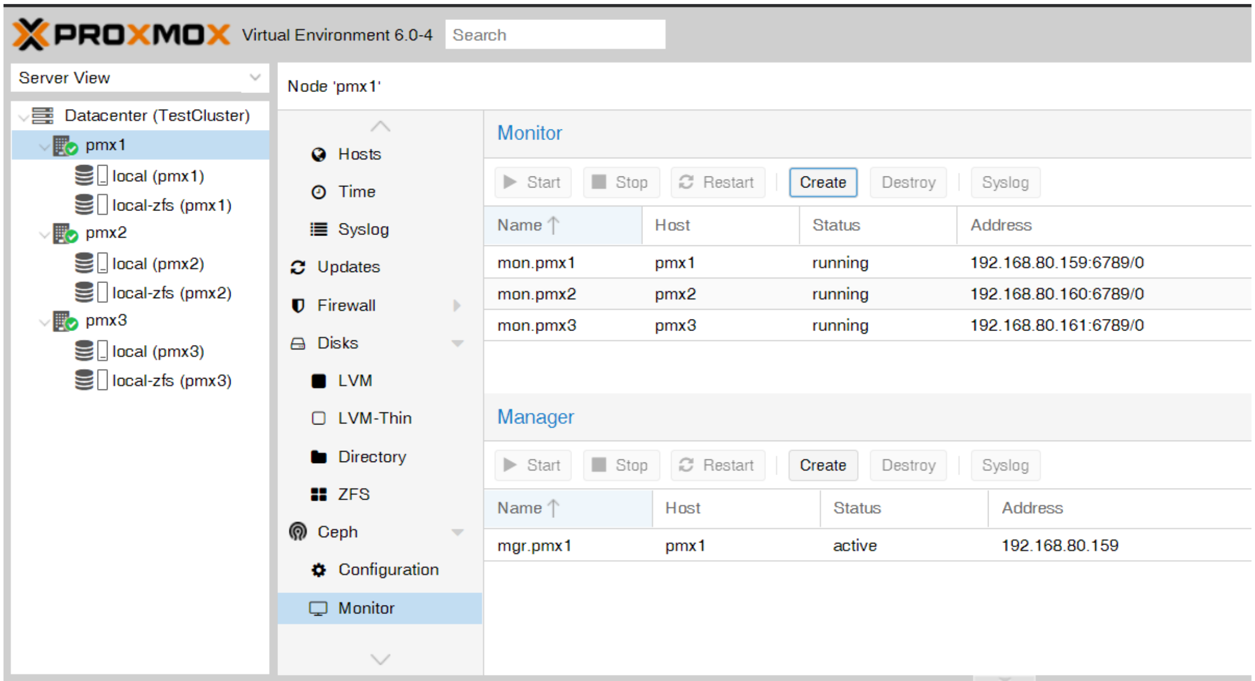
The basic installation and configuration is completed, depending on your setup some of the following steps are required to start using Ceph:

1. Install Ceph on other nodes
2. Create additional Ceph Monitors
3. Create Ceph OSDs
4. Create Ceph Pools

To learn more click on the help button below.

On the two other nodes, no network configuration is needed, the two nodes will configure themselves automatically.

After all the 3 nodes have Ceph up and running, I will create the Ceph monitors on node 2 and 3.



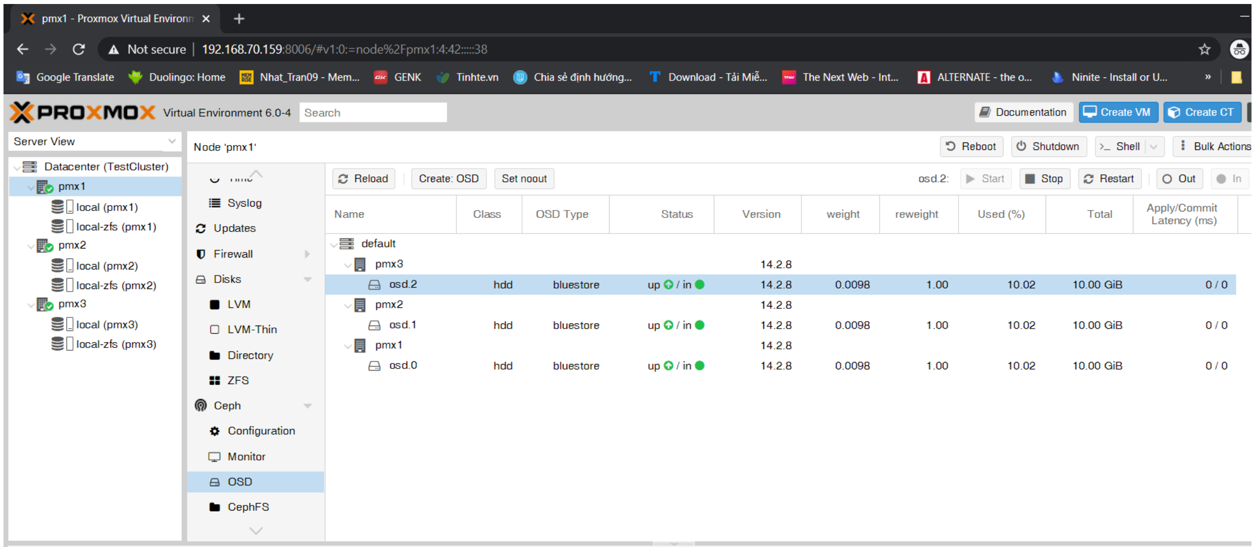
Monitor

Name	Host	Status	Address
mon.pmx1	pmx1	running	192.168.80.159:6789/0
mon.pmx2	pmx2	running	192.168.80.160:6789/0
mon.pmx3	pmx3	running	192.168.80.161:6789/0

Manager

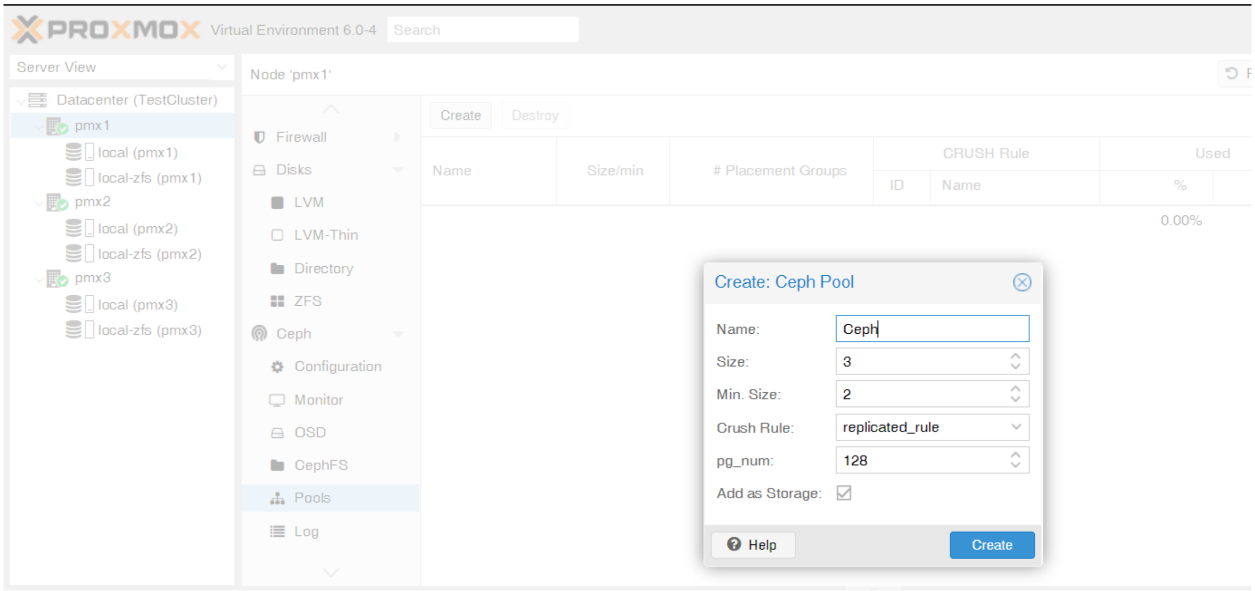
Name	Host	Status	Address
mgr.pmx1	pmx1	active	192.168.80.159

Finally, I will create the OSDs using the extra hard disk has been created. This will be used for the Ceph pool.



Name	Class	OSD Type	Status	Version	weight	reweight	Used (%)	Total	Apply/Commit Latency (ms)
default									
pmx3				14.2.8					
osd.2	hdd	bluestore	up ● / in ●	14.2.8	0.0098	1.00	10.02	10.00 GiB	0 / 0
pmx2				14.2.8					
osd.1	hdd	bluestore	up ● / in ●	14.2.8	0.0098	1.00	10.02	10.00 GiB	0 / 0
pmx1				14.2.8					
osd.0	hdd	bluestore	up ● / in ●	14.2.8	0.0098	1.00	10.02	10.00 GiB	0 / 0

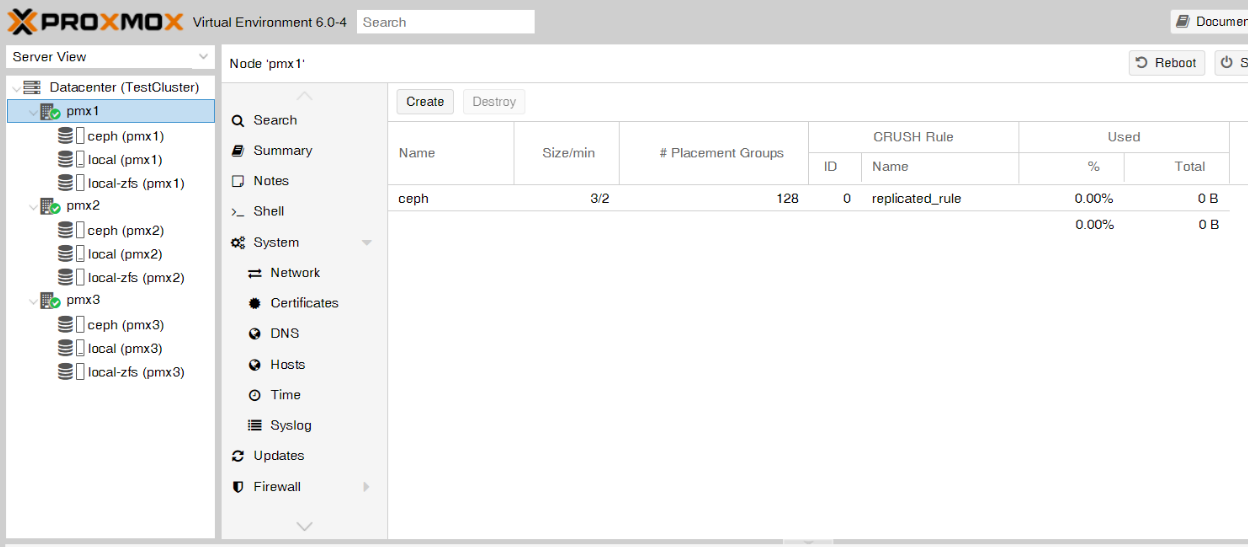
Now I am going to create storage pool, the storage will be available on all nodes



Create: Ceph Pool

Name:	Ceph
Size:	3
Min. Size:	2
Crush Rule:	replicated_rule
pg_num:	128
Add as Storage:	☑

Name	Size/min	# Placement Groups	CRUSH Rule		Used
			ID	Name	%
					0.00%

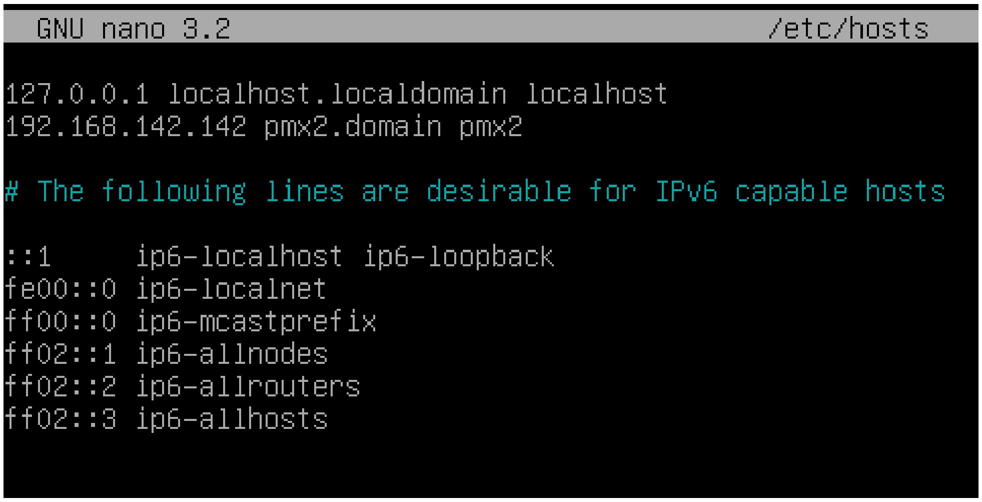


Name	Size/min	# Placement Groups	CRUSH Rule		Used	
			ID	Name	%	Total
ceph	3/2	128	0	replicated_rule	0.00%	0 B
					0.00%	0 B

Command line

Login to each node and config the IP address and hostname with those command line below:

```
nano /etc/hosts
```



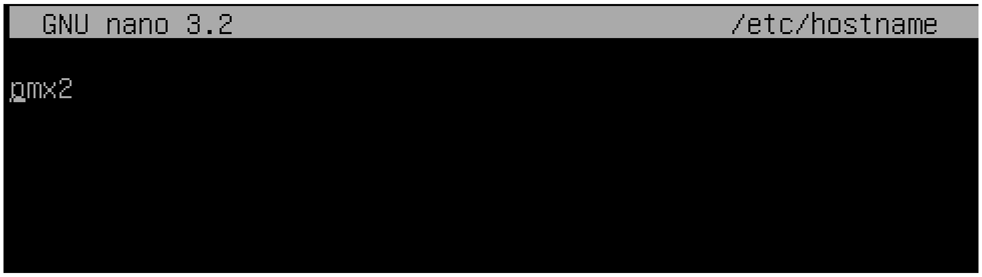
```
  GNU nano 3.2                                  /etc/hosts

127.0.0.1 localhost.localdomain localhost
192.168.142.142 pmx2.domain pmx2

# The following lines are desirable for IPv6 capable hosts

::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

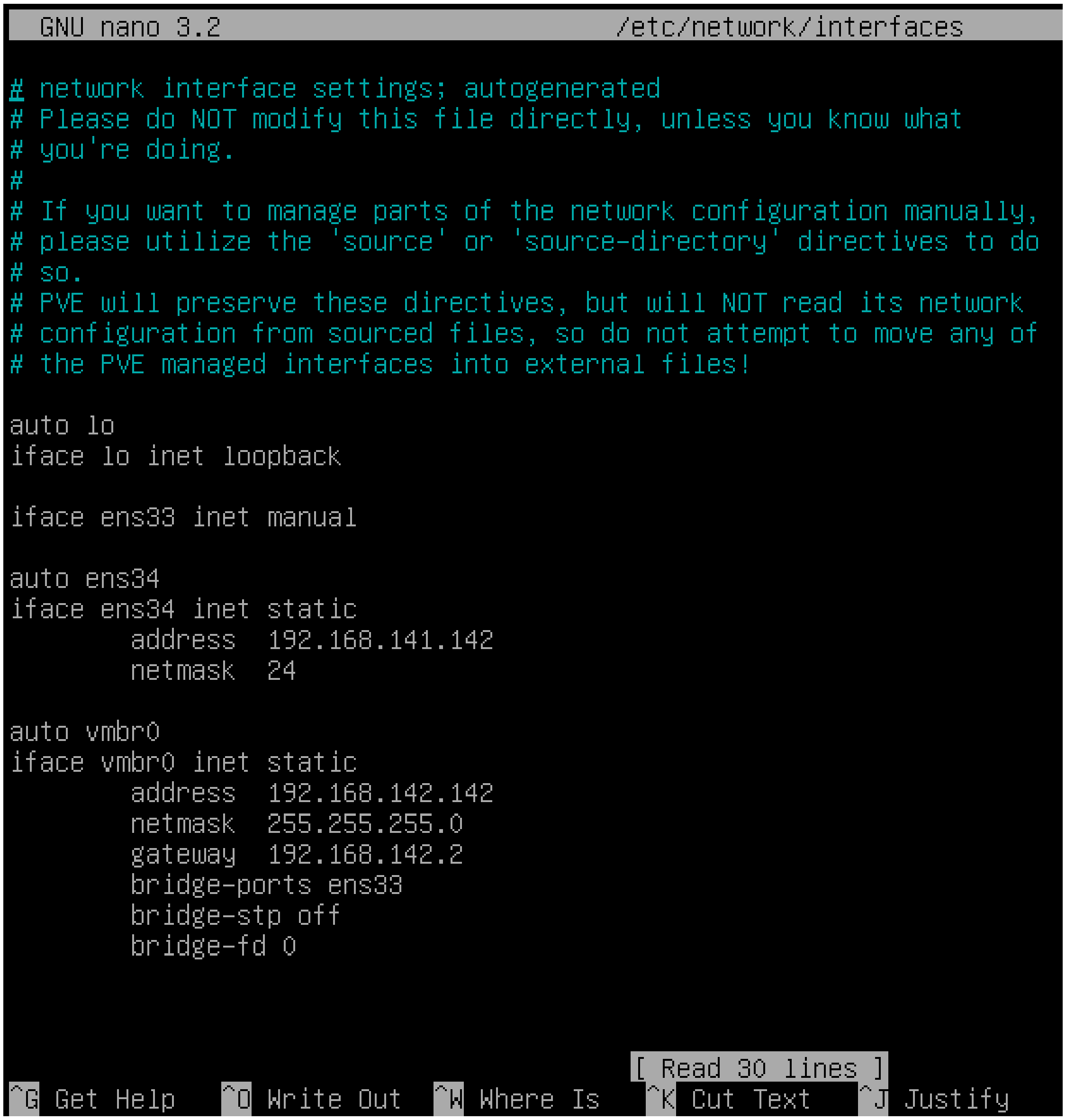
nano /etc/hostname



```
  GNU nano 3.2                              /etc/hostname

pmx2
```

nano /etc/network/interfaces



```
  GNU nano 3.2                                      /etc/network/interfaces

# network interface settings; autogenerated
# Please do NOT modify this file directly, unless you know what
# you're doing.
#
# If you want to manage parts of the network configuration manually,
# please utilize the 'source' or 'source-directory' directives to do
# so.
# PVE will preserve these directives, but will NOT read its network
# configuration from sourced files, so do not attempt to move any of
# the PVE managed interfaces into external files!

auto lo
iface lo inet loopback

iface ens33 inet manual

auto ens34
iface ens34 inet static
        address  192.168.141.142
        netmask  24

auto vmbr0
iface vmbr0 inet static
        address  192.168.142.142
        netmask  255.255.255.0
        gateway  192.168.142.2
        bridge-ports ens33
        bridge-stp off
        bridge-fd 0



                                           [ Read 30 lines ]
^G Get Help     ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify
```

***If I clone node before editing the NIC, it will cause an error like:**

```
Error: Driver 'pcspkr' is already registered, aborting...
```

Workaround solution: Just to follow up, login to the command line as root via console or SSH and run this single line of code to remove that error message:

```
# echo "blacklist pcspkr" > /etc/modprobe.d/blacklist-pcspkr.conf
```

Ceph NOTE: https://github.com/nhanhoadocs/ghichep/wiki/Ghi-ch%C3%A9p-v%E1%BB%81-CEPH

Appendix B: Search tree table

Please find the research article link below:

https://www.dropbox.com/sh/ga3ixgacct1x1o6/AAAQAOK4Bhw6Q_ObdSVCB0hQa?dl=0

Databases	Term	Result
Google Scholar	Cloud computing	Research article: Cloud computing và OpenStack. Authors: Chi Le, Tung Nguyen. Research paper: The NIST Definition of Cloud Computing. Authors: Peter Mell, Timothy Grance. Special Publication 800-145
Google Scholar	Type-1 hypervisors	Research article: Evaluation of type-1 hypervisors on desktop-class virtualization hosts. Authors: Duarte Pousa and José Rufino. IADIS International Journal on Computer Science and Information Systems. Vol. 12, No. 2, pp. 86-101 ISSN: 1646-3692
Google Scholar	Hyper-V ESXi Xen	Research article: Benchmarking the Performance of Microsoft Hyper-V, VMware ESXi, Xen Hypervisors. Authors: Hasan Fayyad-Kazan, Luc Perneel, Martin Timmerman. Journal of Emerging Trends in Computing and Information Sciences. Vol. 4, No. 12, December 2013 ISSN 2079-8407
Google Scholar	Comparison of commercial hypervisors	Research article: A Performance Comparison of Commercial Hypervisors. Author: XenSource, Inc.
Google Scholar	Comparison of open-source cloud frameworks	Research article: A Comparison and Critique of Eucalyptus, OpenNebula and Nimbus. Authors: Peter Sempolinski and Douglas Thain. University of Notre Dame

Google Scholar	Open-source cloud	White paper: Ubuntu Cloud: Technologies for future-thinking companies. Authors: Canonical, 2011.
Google Scholar	OpenStack	Research article: Open-Source Solution for Cloud Computing Platform Using OpenStack. Authors: Rakesh Kumar, Neha Gupta, Shilpi Charu, Kanishk Jain, Sunil Kumar Jangir. International Journal of Computer Science and Mobile Computing, Vol.3 Issue.5, May- 2014, pg. 89-98 Research article: OpenStack Toward an Open-Source Solution for Cloud Computing. Authors: Omar Sefraoui, Mohammed Aissaoui, Mohsine Eleuldj. International Journal of Computer Applications (0975 - 8887) Volume 55 - No. 03, October 2012.
E-book	Proxmox VE/ Proxmox Virtual Environment	Book: Mastering Proxmox 3rd Edition. Authors: Wasim Ahmed. Copyright © 2017 Packt Publishing. Book: Proxmox Cookbook. Authors: Wasim Ahmed. Copyright © 2015 Packt Publishing. Book: Proxmox VE Administration Guide. Authors: Proxmox Server Solutions Gmbh. Release 6.0. July 15, 2019. White Paper: Proxmox VE Administration Guide. Authors: Proxmox Server Solutions Gmbh.
E-book	Ceph storage	Book: Mastering Ceph. Authors: Nick Fisk. Copyright © 2017 Packt Publishing. Book: Learning Ceph second edition. Authors: Anthony D'Atri, Vaibhav Bhembre, and Karan Singh.

		Copyright © 2017 Packt Publishing.
Google Scholar	Security and privacy in cloud computing	Research article: Guidelines on security and privacy in public cloud computing. Authors: Wayne Jansen and Timothy Grance. Special Publication 800-144. Research article: Research paper: Guidelines on security and privacy in public cloud computing. Authors: V. Shobana, M. Shanmugasundaram.
Saxion	Multi Criteria Analysis	Book: Multi Criteria Analysis. Authors: Guillermo A. Mendoza and Phil Macoun with Ravi Prabhu, Doddy Sukadri, Herry Purnomo and Herlina Hartanto.

Appendix C: Other activities

Besides the graduation assignment which were assigned by Mr. Anh Le. There are various interesting activities I had been experienced (both technical and social aspects) during my project.

Since learning as much as possible is my objective. I sometimes followed my colleagues to the Data Centre to install the server. At the moment, we have deployed five-node Proxmox cluster at Viettel IDC Datacenter for testing purpose. Thanks to TinoHost, I had a chance to participate to Vietnam Web Summit 2019. The event is focusing on Internet Technology, technology trends and technological moves in Vietnam market. Speakers have included C-level management levels, top experts to outstanding faces offering compelling content and inspiration. There were various interesting topics in the event that I think they would be new trend in the near future. As a partner of Vietnam Web Summit, TinoHost have a chance to introduce themselves and enhance the reputation. In fact, a large amount of new customer registered to use our service after this event.

Figure 14: Vietnam Web Summit 2019

LIST OF FIGURES

LIST OF TABLES

BIBLIOGRAPHY

[1] K. Haveman, E. Hageraats and S. van der Linden, "Design Research," Saxion, 2016.

[2] K. Haveman, E. Hageraats and S. van der Linden, "Design Research," Saxion, 2016.

[3] T. G. Peter Mell, "The NIST Definition of Cloud Computing," Special Publication 800-145.

[4] T. Hou, "IaaS vs PaaS vs SaaS Enter the Ecommerce Vernacular: What You Need to Know, Examples & More," [Online]. Available: https://www.bigcommerce.com/blog/saas-vs-paas-vs-iaas/#the-three-types-of-cloud-computing-service-models-explained.

[5] c. ace. [Online]. Available: https://vn.cloud-ace.com/column/saas-paas-iaas-la-gi.

[6] VMWare, "Understanding Full virtualization, Para virtualization and hardware Assist," 2007. [Online]. Available: http://www.vmware.com/files/pdf/VMware_paravirtualization.pdf..

[7] A. F. a. P. Lownds, " Mastering Hyper-V Deployment," no. Wiley Publishing Inc..

[8] Fullvirtualization. [Online]. Available: https://en.wikipedia.org/wiki/Full_virtualization.

[9] Paravirtualization. [Online]. Available: http://en.wikipedia.org/wiki/Paravirtualization..

[10] OS-level_virtualization. [Online]. Available: https://en.wikipedia.org/wiki/OS-level_virtualization.

[11] D. o. ESX. [Online]. Available: https://communities.vmware.com/thread/330666.

[12] Canonical, "Ubuntu Cloud: Technologies," September 2011.

[13] OpenStack, "https://docs.openstack.org," 29 11 2018. [Online]. Available: https://docs.openstack.org/arch-design/design-compute/design-compute-hypervisor.html.

[14] L. Kurth, 20 5 2015. [Online]. Available: https://xenproject.org/2015/05/20/xen-project-now-in-openstack-nova-hypervisor-driver-quality-group-b/.

[15] P. Bernard, IT Servcie Management based on ITIL 2011.

[16] W. Ahmed, Mastering Proxmox third edition, PacktPub Publishing Ltd..

VERSION HISTORY

Version	Date	*Changes*
1.0	20 – 03 – 2020	*Cover design & All content chapters.*
1.1	22 – 03 – 2020	*Merge chapters.*
1.2	23 – 03 – 2020	*Draft version.*
1.3	28 – 03 – 2020	*Edit based on received feedback.*
2.0	06 – 04 – 2020	*Final version.*