

Résumé Journalier

Joffrey Hérard

4 avril 2017

1 Les VMs

Liste de mes connaissances des différentes technologies de virtualisation

- VirtualBox
- VMWARE
- LXC
- Docker
- QEMU
- KVM
- HyperV
- Proxmox VE
- Oracle VM Server
- Hyper-V Containers

Réorganisation des différentes VM par tableau :

TABLE 1 – Différents services de Virtualisations	
Conteneurs	Hyperviseurs Types 1 et 2
LXC	Virtualbox
Docker	VMWare
Vagrant	Qemu
	KVM
Hyper-V Containers	Proxmox(Basé sur KVM)
	Oracle VM Server
	Hyper-V

Quoi garder ? Choix sur les différentes VM :

LXC ✓	Docker ✓	HyperV ✗
VMWare ✗	VirtualBox ✓	Qemu ✗
Oracle VM Server ✗	Proxmox ✓	KVM ✗
Hyper-V Containers ✗		

Justification des choix cas par cas :

Postulat 1 *LXC : Gestionnaire de Conteneurs Linux, utilisé aussi souvent que Docker (NB : Docker basé sur LXC)*

Postulat 2 *Docker : Gestionnaire de Conteneurs, que j'ai utilisé très souvent. (NB : Docker basé sur LXC)*

Postulat 3 *VirtualBox : Utilisé de manière courante aussi, de plus c'est un Hyperviseur classique de Type 2.*

Postulat 4 *Proxmox VE : Utilisé d'un point de vue utilisateurs, mais il complète la liste en étant un Hyperviseur de type 1.*

Postulat 5 *VMware : Je ne sais pas si l'on a une licence à notre disposition ?*

Postulat 6 *Qemu : Utilisé quelque fois cependant, basé sur KVM(Comme Proxmox VE) et XEN(Connu que de nom).*

Postulat 7 *Oracle VM Server : Jamais abordé ..et je ne sais pas si l'on a une licence à notre disposition ?*

Postulat 8 *KVM : HyperViseur de type 1. Proxmox étant basé sur KVM et étant lui aussi un Hyperviseur de Type 1,*

Postulat 9 *Hyper-V Containers : Jamais abordé .. ne connaissait pas son existence avant de faire de la recherche sur le sujet .*

Postulat 10 *Hyper-V : Jamais abordé.*

2 Les tests

Test de performance Les informations recueillies concernent les temps de réponse utilisateurs, les temps de réponse réseau et les temps de traitement d'une requête.

Test de montée en charge Il s'agit d'un test au cours duquel on va simuler un nombre d'utilisateurs sans cesse croissant de manière à déterminer quelle charge limite le système est capable de supporter

Test de dégradations Ce test peut tenir compte ou non de la cadence des itérations, la représentativité en termes d'utilisateurs simultanés vs. sessions simultanées n'étant pas ici un objectif obligatoire, s'agissant ici plutôt de déterminer les points de contention générés par chaque scénario fonctionnel.

Test de robustesse il s'agit de tests au cours desquels on va simuler une charge importante d'utilisateurs sur une durée relativement longue, pour voir si le système testé est capable de supporter une activité intense sur une longue période sans dégradations des performances et des ressources applicatives ou système.

Test aux limites Proche du test de capacité,

Benchmark

3 Retour sur la liste des choses a faire

Retour sur le point 3 En réalité, j'ai mis les use case a cette place, mais sur le schema de la page suivante on voit que il ce fait en "parallèle ".Mais effectivement on pourrais le mettre en première positions puis faire l'ensemble des autres tache à faire.

Retour sur le point 6 Parce que à ma connaissance, il y a probablement plusieurs(A entendre différent services)déjà installer, et sachant que l'on doit faire des test, il est nécessaire de "nettoyer la machine" afin de n'avoir que ce qui est utile a nos besoins, et pas avoir des Démons qui s'exécutent et prennent donc des ressources.

Retour sur le point 8 a

Retour sur le point 8 c

Références

- [1] Documentation Docker <https://docs.docker.com/>.
- [2] Hyper-V Containers <https://docs.microsoft.com/en-us/virtualization/windowscontainers>.
- [3] Linux Containers <https://linuxcontainers.org/fr/>.
- [4] Virtual Box <https://www.virtualbox.org/>.
- [5] Proxmox VE <https://www.proxmox.com/en/>.
- [6] VmWare <http://www.vmware.com/fr.html>.
- [7] QEMU <http://www.qemu-project.org/>.
- [8] Hyper-V <https://www.microsoft.com/fr-fr/cloud-platform/server-virtualization>.
- [9] KVM <https://www.linux-kvm.org/>.
- [10] Wikipedia sur les Tests https://fr.wikipedia.org/wiki/Test_de_performance.
- [11] VM Mark <https://www.vmmark.com/products/vmark.html>.