

# **Les fondamentaux de la virtualisation**

## **Partie III**

## Types de virtualisation

- Emulation
- Virtualisation des serveurs : Hyperviseur Type 1 et Type 2
- Virtualisation complète (Full Virtualization)
- Para-virtualisation (Paravirtualization)
- Virtualisation du système d'exploitation
- Virtualisation des applications

## L'émulation (Emulation) (1/5)

➤ **L'émulation** est une technique qui permet de simuler le **fonctionnement de n'importe quel ordinateur (processeur et matériel)** sur un autre **ordinateur même si celui-ci est technologiquement différent** (architecture matérielle complètement différente de la machine physique hôte)

càd **reproduire de manière logicielle** l'ensemble des **ressources matérielle** d'une machine, ce qui demande **énormément de ressources** pour faire tourner de manière fluide un système basé sur une architecture différente, sans pénaliser le système d'exploitation lui-même.

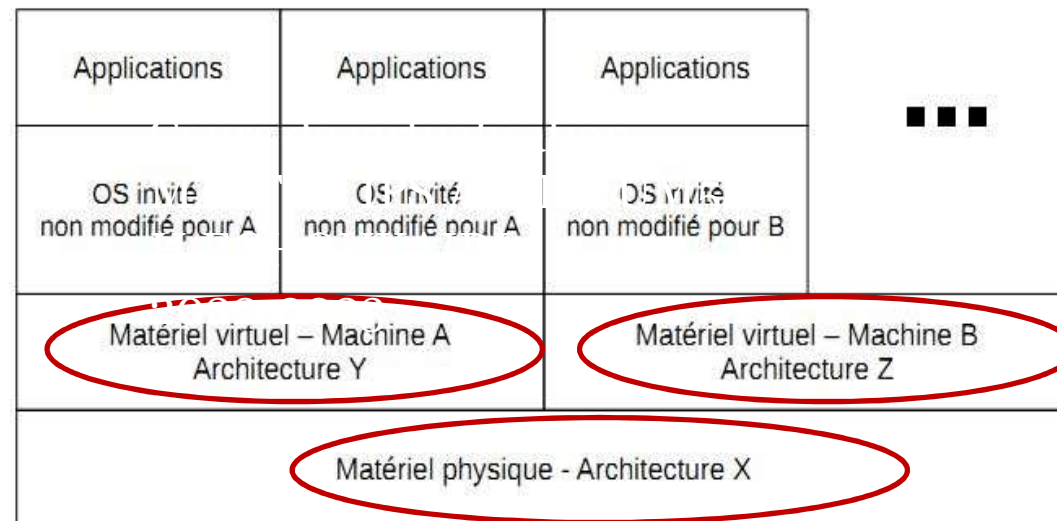
➤ **Imitation du comportement physique d'un matériel par un logiciel**, et ne pas la confondre avec la simulation, laquelle vise à imiter un modèle abstrait.

❑ L'émulation permet **non pas de modéliser**, mais bel et bien de **reproduire à l'identique** le comportement d'un logiciel et son architecture matérielle.

## L'émulation (Emulation) (2/5)

➤ Isoler les systèmes d'exploitation invités et cohabiter plusieurs systèmes d'exploitation hétérogènes sur une même **machine hôte**. Cela permet d'exécuter des applications prévues pour d'autres architectures.

Par contre, cette solution est **coûteuse en performance** du fait de l'émulation du processeur.



- ☐ L'émulateur de machines virtuel fournit une architecture virtuelle qui peut ne pas être différente de l'architecture du système hôte
- ☐ OS invités s'exécute sans modification sur le matériel virtuel

Source : Voir Références

## Emulation : Exemples (3/5)

### ❑ Exemples :



1) Emulation d'anciennes consoles sur nos nouveaux ordinateurs tel que **game boy**.

2) VMware Network Adapter VMnet1 (**voir Démo**)

1- VMware Workstation /VM / Settings/ Network adapter

2-Panneau configuration / Réseau et Internet / connexions réseau /VMware Network Adapter VMnet1

3) Émulateur de terminal (lien avec la virtualisation)

Utiliser des commandes ou des logiciels Windows sur une machine UNIX ou vice-versa.

❑ La simulation numérique est l'un des **outils permettant de simuler des phénomènes réels**.

❑ Simulation informatique désigne un **procédé** selon lequel on exécute un programme informatique sur un ordinateur en vue de **simuler par exemple un phénomène physique complexe** (NS2, NS3, NetSim,...)

## Emulation : comparaison (4/5)

Virtualization	Emulation
Hardware can accessed directly	You'd need a software connector to access hardware bridge
the virtual machine runs code directly, which available in different languages.	Emulator requires an interpreter to translates the source code and converts it to the host system's readable format.
VMs make use of CPU.	Emulators do not rely (ne repose pas) on CPU
Virtualization puts a layer between hardware physically to control access to it.	
VM are relatively faster in its operations	Emulators are relatively slower
VM solution is costlier and more complex	Emulation is comparatively cheaper
It provides more better backup solutions	Emulation falls short of (sans répondre) virtualization as far as backup are recovery is considered

Source : Voir Références

## L'émulation (Emulation) : Acteurs (5/5)

➤ **QEMU** : émulateur de plateformes **x86 (32bits)**, **Sparc**,.

Il s'agit d'un logiciel libre de machine virtuelle qui a la possibilité **d'émuler plusieurs type de processeurs**.

➤ **KVM (Kernel-based Virtual Machine)** : version modifiée de QEMU tirant parti des instructions Intel VT et AMD-V. Il s'agit d'un **hyperviseur libre de type I** pour **Linux**.

➤ **BOCHS** : émulateur libre très complet de **l'architecture PC traditionnelle** x86 (processeur Intel), émulant un processeur entier dans le logiciel même quand il est en cours d'exécution sur un processeur physique compatible.

➤ **Microsoft VirtualPC et Microsoft VirtualServer** : propriétaire, émulateur de plateforme x86

➤ **Virtual Box** : Est un logiciel sous licence GNU GPL depuis janvier 2007 développé par SUN .Il gère tous les OS tels que Windows, Linux ou Mac OS

➤ **VMWare** : propriétaire, émulateur de plateforme x86 (produits VMware Server, VMware Player et VMware Workstation)

Source : Voir Références

# Anneaux de protection

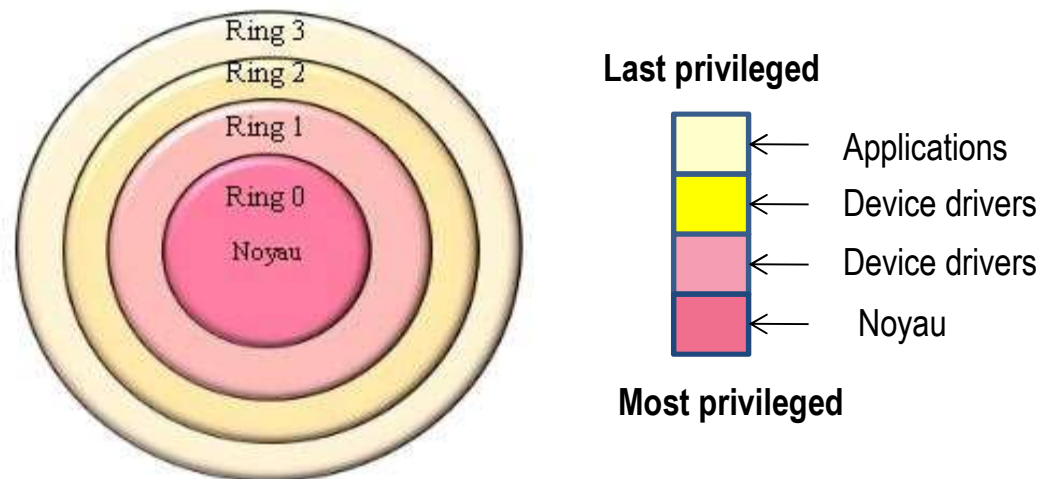
- ❑ Les anneaux de protection (**protection rings**) offrent **plusieurs niveaux de privilèges**, du plus faible au plus élevé
- ❑ Ils sont mis en œuvre au **sein de la mémoire vive**
- ❑ Quand un programme s'exécute, le système d'exploitation lui attribue une **zone de mémoire vive** qui est affectée à un anneau de protection.
- ❑ L'intérêt est de faire une **séparation entre les programmes sensibles**, comme le noyau du système, et les programmes du ou des utilisateurs.
- ❑ Un programme contenu dans une zone mémoire qui est affectée à l'anneau 3 **ne pourra pas accéder** à une zone mémoire affectée à l'anneau 0.
- ❑ L'organisation des anneaux de protection **varie** si nous avons d'une part une installation **physique ou virtualisée**. Et d'autre part, si **l'architecture est 32 bits ou 64 bits**.



## Virtualisation et Anneaux de protections (privilèges) au Processeur X86 (1/2)

Afin de comprendre le fonctionnement de ces technologies, il est important de savoir que **l'architecture des processeurs** de la famille x86 implémentent **quatre anneaux de protections (niveaux d'exécutions, ou Rings)** qui :

- Définissent les **privilèges d'exécution des programmes** (Ring de 0 à 3)
- Sont arrangés dans une hiérarchie allant du **plus privilégié au moins privilégié** (figure ci-dessous).



**NB :** Le programme installé sur un **niveau bas** exerce **plus de contrôle sur le système**

Source : Voir Références

## Anneaux de protections (privilèges) : processeur X86

On distingue 4 anneaux de protections (privilèges) :

**Ring 0** : Anneau de protection (niveau le plus bas) ayant le **plus de privilèges (plus sécurisé)**, et sous lequel fonctionne le **cœur du système d'exploitation (Kernel mode)**

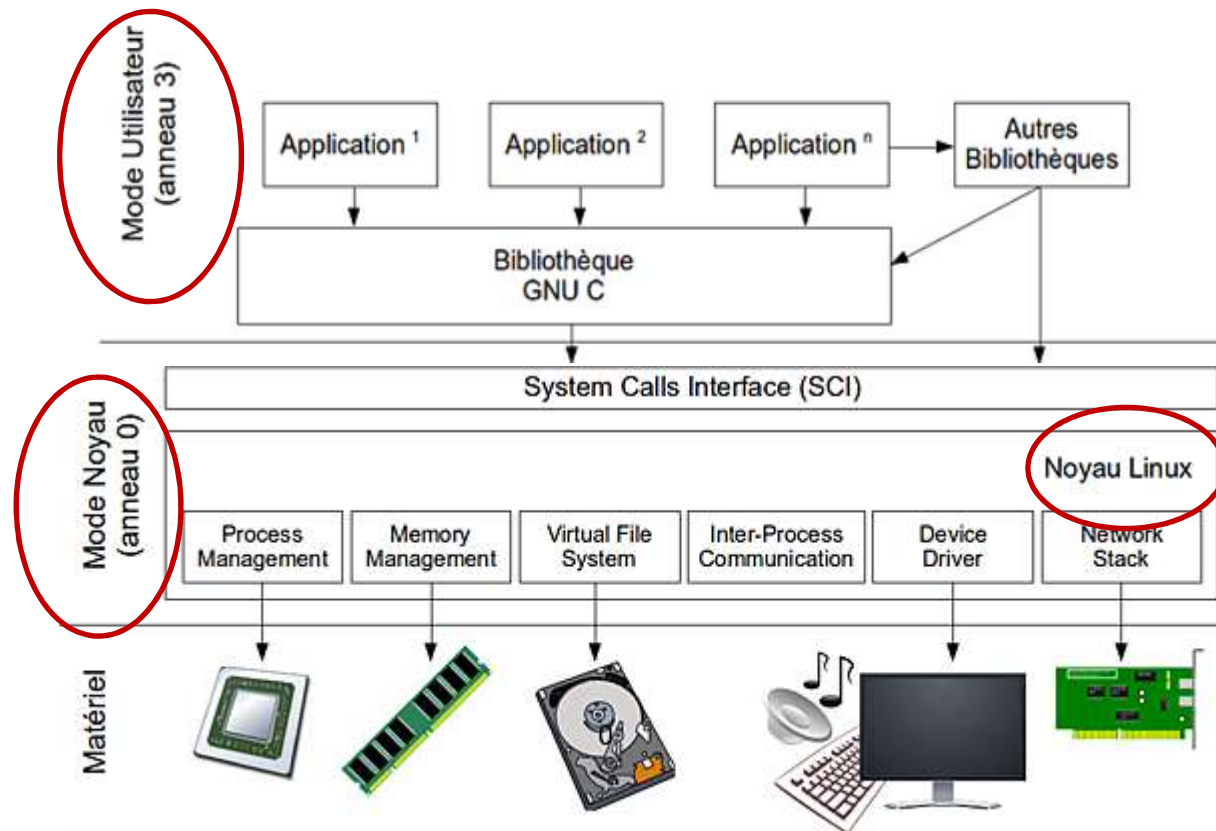
**Ring 1** : non utilisé, définit des privilèges de moindre importance

**Ring 2** : non utilisé, définit des privilèges de moindre importance

**Ring 3** : est le niveau le plus élevé ayant **moins de privilèges (moins sécurisé)**, il représente l'espace utilisateur sous lequel tourne les applications . Ces applications ne peuvent pas modifier ce qui s'exécute sur des Rings inférieurs.

# Anneaux de protection et Noyau

2) Les applications communiquent avec le noyau par le biais des appels système (System Calls ou syscalls)



1) noyau Linux s'exécute dans l'anneau de protection 0

**Exemple:** Une application ne peut pas arrêter l'OS, alors que l'OS peut arrêter une application.

## Virtualisation des serveurs : Hyperviseur Type 1 et Type 2

- Un **hyperviseur** est une approche **bare-metal ( métal nu)** de la virtualisation.
  - Bare-metal fait référence au **matériel du serveur sans système d'exploitation installé.**
- Il s'agit d'un outil de virtualisation qui **fait fonctionner un ou plusieurs système d'exploitation sur une même machine physique en même temps.**
- Avec la virtualisation, le **système d'exploitation invité** accède à **l'architecture matérielle** sous-jacente par l'intermédiaire d'un **noyau système très léger** nommé **hyperviseur**.

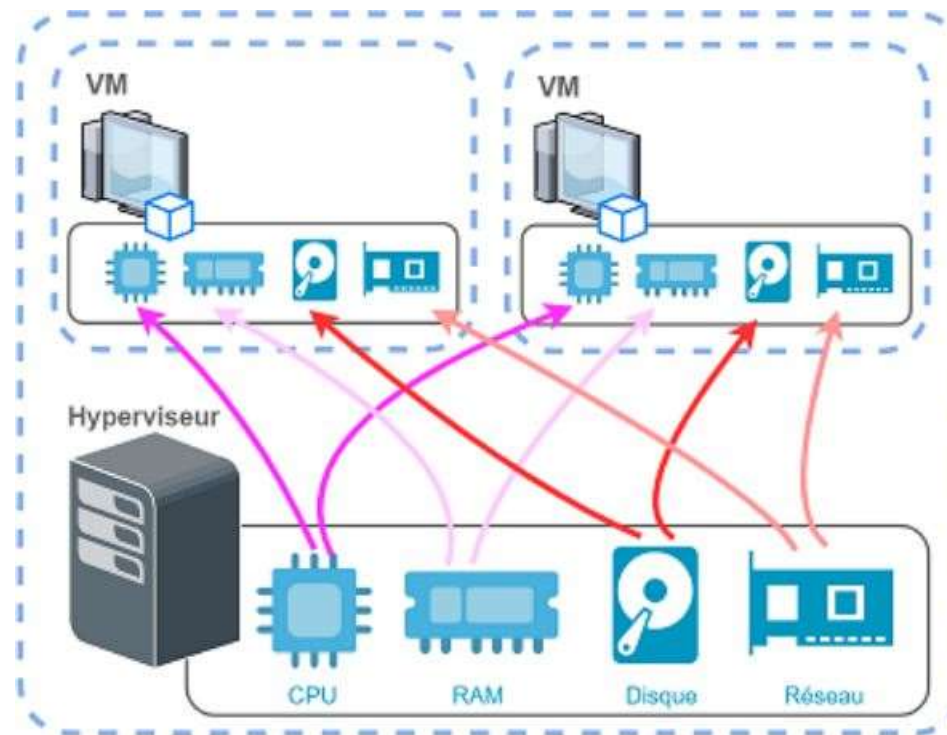
**Hyperviseur = mini système installé sur le système hôte qui gère le matériel**

- Le système hôte était le **seul à avoir un accès direct au matériel.** Avec un hyperviseur, il partage cet accès avec les systèmes invités.

Source : Voir Références

## Virtualisation des serveurs : Hyperviseur Type 1 et Type 2

➤ Avec l'hyperviseur, chaque VM possède ces **propres ressources** en termes de : disque dur, mémoire, processeur et périphériques à la différence près que tout cela sera **virtuel**



Création des ressources virtuelles propres à chaque VM

Source : Voir Références

## Virtualisation des serveurs : Hyperviseur Type 1 et Type 2

➤ Plusieurs systèmes d'exploitation peuvent être exécutés quasiment directement **au-dessus** du matériel, **sans être dépendant d'un OS hôte**.

➤ **Rôles d'un hyperviseur sont :**

- assurer le contrôle du processeur et des ressources de la machine Hôte;
- allouer à chaque VM les ressources dont elle a besoin;
- s'assurer que les VMs n'interfèrent pas l'une avec l'autre.

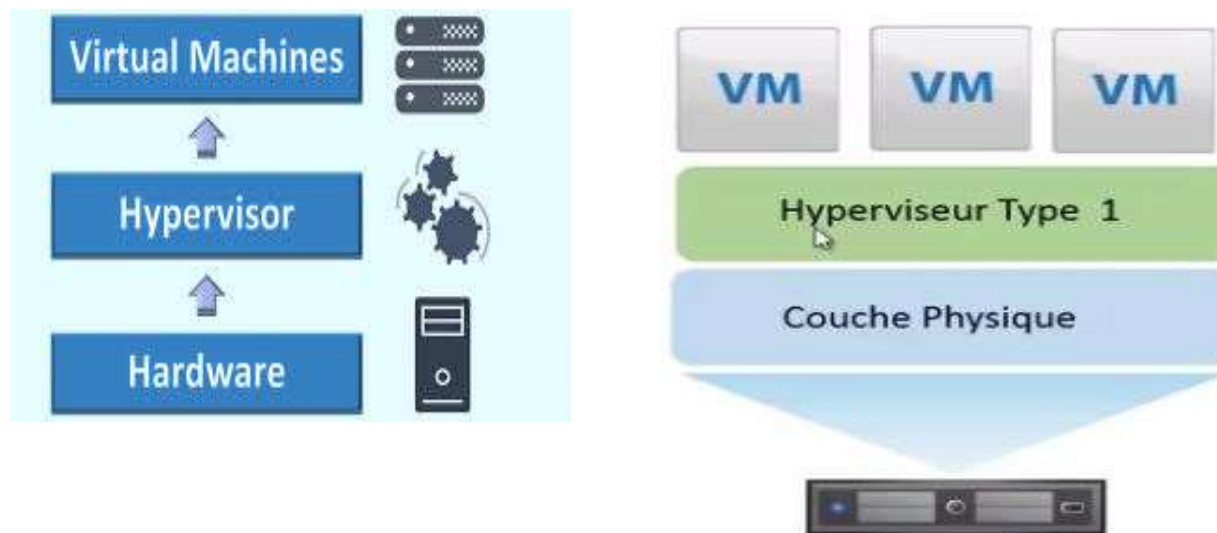
➤ Lorsque la couche de la virtualisation est installée, elle utilise un hyperviseur avec une architecture **type 1 ou en type 2**

**Le type 1, dit native, et le type 2, dit logiciel**

Source : Voir Références

# Hyperviseur Type 1

- Un **hyperviseur de type 1**, ou natif, est un logiciel qui **s'exécute directement sur une plateforme matérielle** (serveur Bar-Metal ) **sans nécessité d'un système exploitation** (sans OS intermédiaire)



Hyperviseur de Type 1

- Ce système est **allégé et optimisé** de manière à se « concentrer » **de plus** sur la **gestion des systèmes d'exploitation invités** c'est-à-dire ceux utilisés par les machines virtuelles qu'ils contiennent
- >> Outil de contrôle non seulement du matériel mais aussi d'un ou plusieurs systèmes d'exploitation invités

Source : Voir Références

## Hyperviseur Type 1

- Hyperviseur Type 1 se comporte comme **un mini OS**
- **Au démarrage** de la machine physique :
  - L'hyperviseur prend directement le contrôle du matériel.
  - Il alloue l'intégralité des ressources aux machines hébergées.
- L'installation l'hyperviseur type 1 (ESXi) **efface complètement le système d'exploitation et les données avant de les remplacer.**
- Hyperviseur Type 1 est utilisé pour les **grosses architectures réseaux** d'entreprise qui nécessitent des **optimisations de coûts et de maintenance.**



# Hyperviseur Type 1

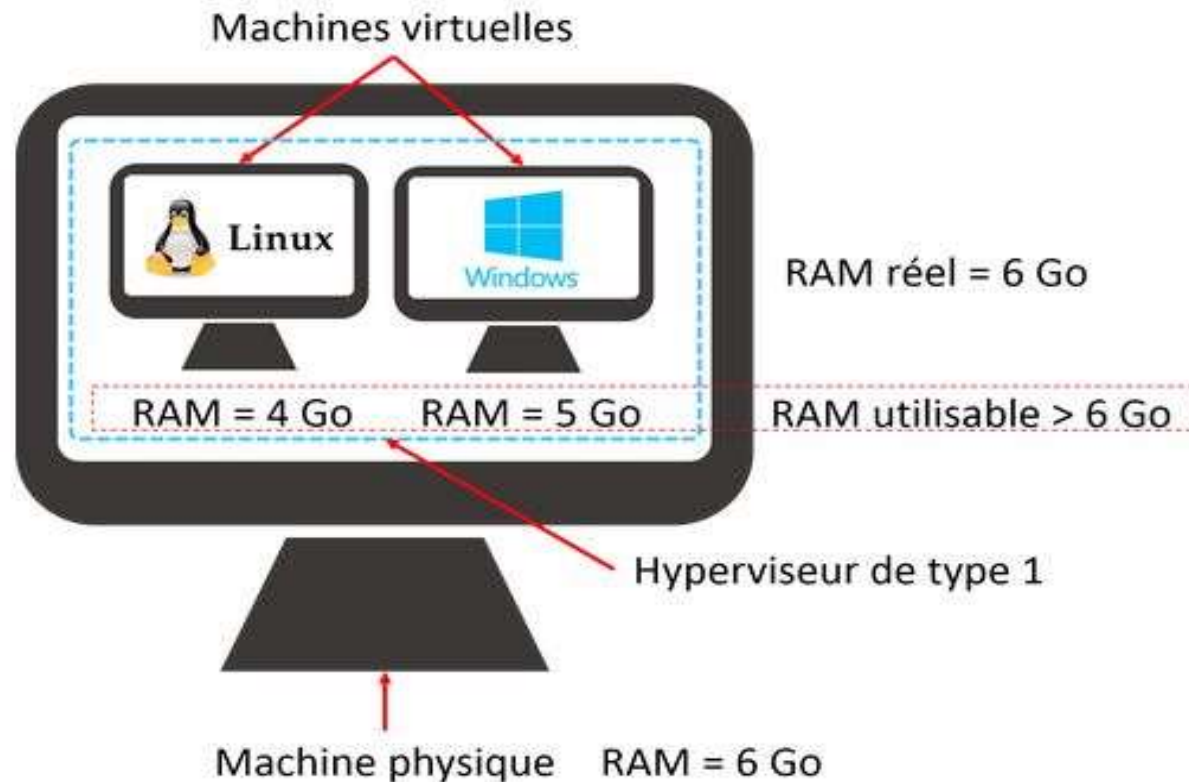
## Avantages :

- ☐ **Allouement de la quasi-totalité des ressources** disponibles aux machines virtuelles, car l'hyperviseur est directement relié à la couche matérielle.
- ☐ Utilisé pour **grosses architectures réseaux d'entreprise**, qui nécessitent des optimisations de coûts et de maintenance, tout en améliorant la robustesse face aux pannes
- ☐ Utile dans la virtualisation de **gros serveurs (type Exchange)** qui demandent de ressources pour fonctionner correctement

## Inconvénients :

Lors de l'exécution de plusieurs instances sur un serveur, le total de **RAM alloué aux machines virtuelles** peut être **supérieur à la capacité de mémoire physique totale du matériel** sous-jacent, ce qui permet une **allocation excessive et dynamique de la capacité du RAM**.

# Hyperviseur Type 1



Dans le cas de la virtualisation beaucoup de machines, il est nécessaire de :

- **Disposer d'une machine physique disposant d'une puissance équivalente à l'intégralité des machines** virtualisées
- Prévoir **plusieurs machines physiques avec d'autres hyperviseurs**.

Source : Voir Références

## Acteurs de la virtualisation basée sur l'hyperviseur Type1

Plusieurs solutions de virtualisation ont été développées pour fonctionner par l'hyperviseur Type1 telles que :

**Xen, Hyper-V et vSphere :**

### **VMware : ESX , ESXi (vSphere)**

- La solution de VMWare prend en charge (dans sa version gratuite) :
  - Un nombre **illimité de CPU physique**, limité à **480 CPU logiques**
  - Jusqu'à **12To** de mémoire vive
  - Pas de limite de machines virtuelles
- Le minimum requis pour faire fonctionner VMWare vSphere Hypervisor est :
  - Un processeur **x64** (avec 2 Cores minimum)
  - **4Go** de mémoire vive
  - Taille de l'hyperviseur VMware est de quelques centaines de Mo sur le disque dur.

Source : Voir Références

## Acteurs de virtualisation basé sur l'hyperviseur Type1

- **Hyper-V (Microsoft)** : couche qui s'ajoute **au dessus** du système exploitation Windows (**Type 1**)
  - La solution de Microsoft **Hyper-V Server 2012 R2** prend en charge :
    - Les systèmes avec **320 CPU** logiques au maximum
    - Jusqu'à **4 To** de mémoire vive
    - Jusqu'à **1024** machines virtuelles
    - Chaque VM avec au maximum **64 processeurs virtuels** et **1 To** de mémoire virtuelle.
  - Le minimum requis pour faire fonctionner Hyper-V server 2012 R2 est :
    - Un processeur **x64** prenant en charge **Intel-VT** ou **AMD-v** cadencé à **1.4 Ghz**
    - **512 Mo** de RAM
  
- **Citrix XenServer (Libre)** : Il est fini, c'était un grand acteur
  - La solution Citrix prend en charge :
    - Jusqu'à **160 CPU** logiques
    - Jusqu'à **1To** de RAM
    - **650 VM** sous Linux, **500 VM** sous Windows
  - Le minimum requis pour faire fonctionner XenServer est :
    - Un processeur **x64** ou **x86** supportant **Intel-VT** ou **AMD-V**, cadencé à **1.5Ghz**
    - **2Go** de mémoire vive

Source : Voir Références

# TRAVAUX PRATIQUES

## **TP N°1 : INSTALLER ET CONFIGURER UNE MACHINE VIRTUELLE PERSONNALISEE SUR VMWARE WORKSTATION Pro**

Source : Voir Références