

Conteneurisation vs Virtualisation

Conteneurisation

La conteneurisation permet d'exécuter des applications dans des environnements isolés appelés conteneurs, qui partagent le même noyau d'exploitation tout en étant indépendants les uns des autres.

Enjeux

- **Portabilité** : Les applications conteneurisées peuvent être déplacées et exécutées sur différentes plateformes (ordinateurs locaux, cloud, serveurs) sans modification.
- **Microservices** : La conteneurisation facilite la mise en œuvre d'architectures basées sur des microservices, où chaque service est isolé dans un conteneur.
- **DevOps et CI/CD** : L'intégration de la conteneurisation dans les pipelines DevOps améliore la continuité du développement, des tests, et du déploiement (via des outils comme Docker et Kubernetes).

Avantages

- Légèreté : **Contrairement à la virtualisation complète, les conteneurs n'incluent pas de système d'exploitation complet, ce qui les rend plus légers et plus rapides à démarrer.** * **Isolation** des applications : Chaque conteneur est isolé, minimisant l'impact d'une défaillance d'une application sur d'autres applications ou services.
- Scalabilité : **Facilité de redimensionner les services et d'augmenter la capacité rapidement en fonction des besoins (grâce à des orchestrateurs comme Kubernetes).**
- * **Efficacité des ressources** : Comme les conteneurs partagent le même noyau, ils consomment moins de ressources que les machines virtuelles (VM).

Limites

Sécurité : Le partage du noyau de l'hôte peut exposer les conteneurs à des vulnérabilités si une faille est exploitée dans le système d'exploitation sous-jacent.

Complexité de l'orchestration : À grande échelle, la gestion des conteneurs peut devenir complexe, nécessitant des outils spécialisés (comme Kubernetes ou Docker Swarm) pour orchestrer et automatiser leur gestion.

Moins d'isolation que la virtualisation : Bien que chaque conteneur soit isolé, cette isolation est moindre comparée à celle des VM, ce qui peut poser des problèmes pour des applications exigeantes en matière de sécurité.

Virtualisation

La virtualisation consiste à créer plusieurs machines virtuelles (VM) sur une seule machine physique, chaque VM ayant son propre système d'exploitation et étant complètement isolée des autres.

Enjeux

* **Optimisation de l'infrastructure** : Utiliser plusieurs VM sur un même serveur physique permet d'optimiser l'utilisation des ressources matérielles.
* **Hébergement multi-tenant** : Chaque VM peut héberger un environnement distinct, ce qui est utile dans les environnements de cloud computing.
* **Indépendance de l'hyperviseur** : L'hyperviseur gère la distribution des ressources physiques entre les VM, offrant une isolation totale.

Avantages

- **Isolation forte** : Chaque VM est isolée au niveau du système d'exploitation, offrant ainsi une sécurité et une stabilité supérieures en cas de défaillance ou de cyberattaque sur une VM spécifique.
- **Flexibilité des systèmes d'exploitation** : Les VM peuvent exécuter différents systèmes d'exploitation sur un seul hôte physique (ex. : une VM sous Linux et une autre sous Windows).
- **Sécurité accrue** : Comme chaque VM a son propre système d'exploitation, l'impact d'une attaque est limité à cette VM, sans affecter les autres VM ou l'hôte.

Limites

- **Surcharge de ressources** : Chaque VM nécessite un système d'exploitation complet, ce qui entraîne une consommation de ressources plus élevée (mémoire, processeur) par rapport aux conteneurs.
- **Démarrage plus lent** : Le démarrage d'une VM est plus lent qu'un conteneur car elle doit charger un OS complet.
- **Moins agile** : Comparée à la conteneurisation, la virtualisation est moins adaptée à des architectures modernes comme les microservices, où la rapidité de démarrage et la légèreté des ressources sont cruciales.

Comparaison et Choix

Cas d'usage

La conteneurisation est plus adaptée aux applications modernes, agiles, nécessitant une scalabilité rapide et une gestion optimisée des ressources (idéal pour les microservices et les environnements DevOps). **La virtualisation** est plus appropriée dans des environnements où une isolation forte est nécessaire, ou lorsque plusieurs systèmes d'exploitation doivent être exécutés sur le même matériel.

Performance

- Les conteneurs sont plus légers et démarrent plus rapidement que les VM.
- Les VM offrent une isolation complète mais au prix d'une plus grande consommation de ressources.

Gestion

- Les conteneurs nécessitent des outils d'orchestration (comme Kubernetes) pour être gérés efficacement à grande échelle.
- Les VM, bien que plus lourdes, peuvent être plus faciles à gérer dans des environnements où l'isolation complète est primordiale.

From:

<https://wiki.ox2.fr/> - Ox2



Permanent link:

<https://wiki.ox2.fr/doku.php?id=cesi:grandoral:benchmark:cloudvirtu&rev=1726330322>

Last update: **2024/09/14 18:12**