

POLYTECHNIQUE YAOUNDE

MIGRATION DE MACHINES VIRTUELLES DANS UN ENVIRONNEMENT HETEROGENE

Superviseur : Prof Alain Tchana
TCHUDJING RUBEN
NGUEKENG MARC DONALD
DONGMO VINCESS
NGOM MBOCK MICHEL
TSANGA ZO ONYABA
NNANGE AKUME

30 janvier 2023

Résumé

Dans le cadre de notre projet de virtualisation , il nous a été proposé comme theme " permettre la migration de machines virtuelles dans un environnement heterogene".Ayant deux machines hotes A et B on veut pouvoir extraire une machine virtuelle sur A et la relancer sur la machine B tout en s'assurant que les applications presentes sur la machine virtuelle fonctionnent toujours sur la machine B.AApres une formalisation mathematique du probleme et une identification de celui ci , nous avons constaté que le probleme se trouve aux niveau des features des machines hotes qui sont souvent differents.Comme solution , nous proposons de créer un programme qui s'exécutera sur la machine hôte de départ qui prend en entrée une application et retourne l'ensemble des features du processeur utilisées lors du fonctionnement de la machine virtuelle et de l'application. Ces features seront ensuite utilisées pour la configuration des features du processeur présentées à la machine virtuelle par la machine hôte de destination.

Table des matières

1	PROBLEME	1
1.1	IDENTIFICATION ET FORMALISATION MATHEMATIQUE DU PROBLEME	1
2	SOLUTION DU PROBLEME	2
2.1	PRESENTATION DE LA SOLUTION	2
2.2	Activer la connexion entre le domU et le dom0	2
2.2.1	Configuration au niveau du dom0	3
2.2.2	Configuration au niveau du domU	3
2.3	Connexion au domU à internet	3
2.3.1	Configuration au niveau du dom0	3
2.3.2	Configuration au niveau du domU	3
3	IMPLEMENTATION	4
3.1	Architecture de la solution	4
3.1.1	Module de configuration et de démarrage de la machine virtuelle	4
3.1.2	Module d'extraction des features essentiels au fonctionnement de la machine virtuelle et de l'application	4
3.1.3	Module de configuration de la machine virtuelle pour la migration	4

Chapitre 1

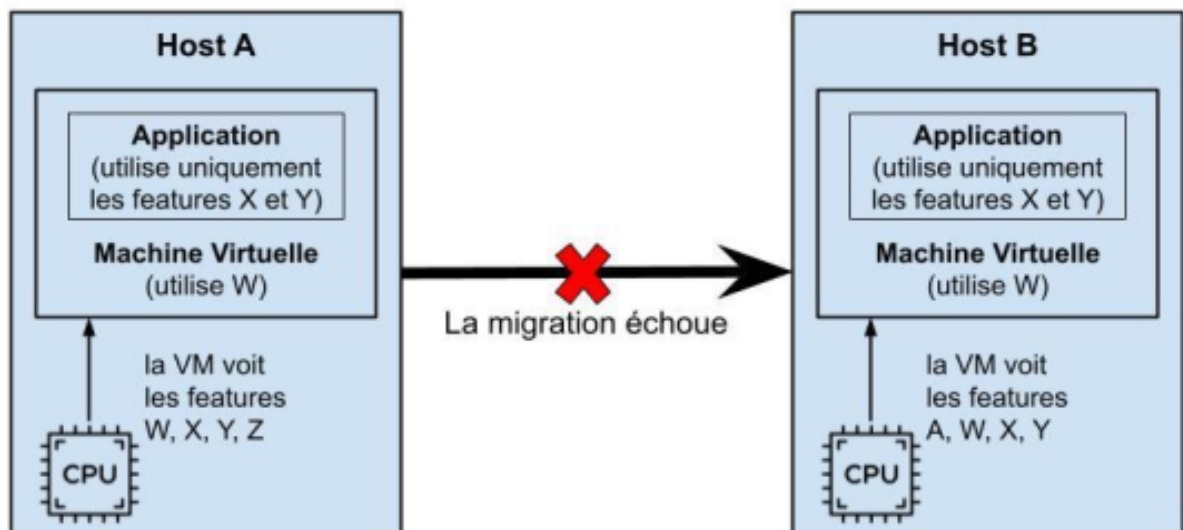
PROBLEME

1.1 IDENTIFICATION ET FORMALISATION MATHEMATIQUE DU PROBLEME

Soient :

- deux machines hotes A et B ou A est la machine de depart et B la machine destinataire
- une machine virtuelle VM s'executant sur la machine de depart A
- une application s'executant sur VM
- une fonction `VMfeature()` qui retourne les features de la machine virtuelle VM lorsque l'application s'exécute
- une fonction `Hostfeature()` qui affiche les features de la machine hote

le probleme se trouve aux niveaux des valeurs renvoyées par la fonction `Hostfeature()`. en effet cette derniere affiche des resultats differents lorsque elle est appliqué aux hotes A et B ($\text{Hostfeature}(A) \neq \text{Hostfeature}(B)$). Et ce probleme survient meme quand les features propres a l'execution de la machine virtuelle c'est a dire lorsque $\text{VMfeature}(C)$ ($\text{Hostfeature}(A)$ et $\text{Hostfeature}(B)$)

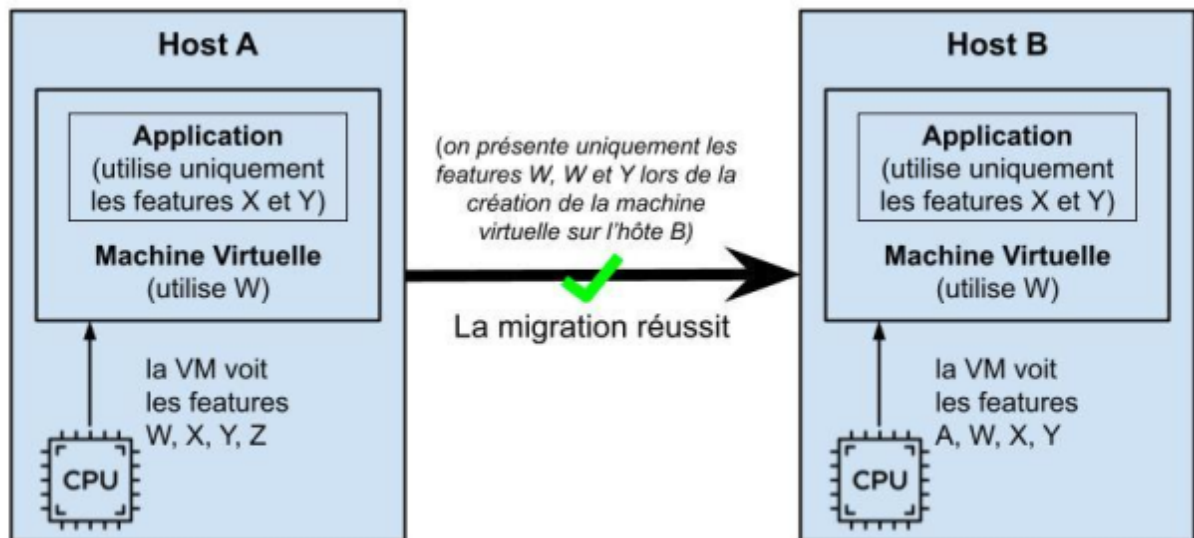


Chapitre 2

SOLUTION DU PROBLEME

2.1 PRESENTATION DE LA SOLUTION

Comme solution nous avons créé un programme qui s'exécutera sur la machine hôte de départ qui retourne l'ensemble des features du processeur utilisées lors du fonctionnement de la machine virtuelle et de l'application. Ces features seront ensuite utilisées pour la configuration des features du processeur présentées à la machine virtuelle par la machine hôte de destination.



voici les grandes etapes de realisation de la solution

- installer xen en suivant les commandes du fichier install-xen
- activer la connexion entre le dom0 et le domU
- installer postgre sur la machine virtuelle et lancer les tests puis recuperer les features
- ecrire un programme capable de tester la pertinence de chaque feature pour avoir uniquement les features utiles

2.2 Acitver la connexion entre le domU et le dom0

L'objectif ici est de créer une connection entre l'hôte (dom0) et les marchines virtuelles (domU), de sorte que le ping de l'un à l'autre marche. Les paramètres de notre environnement sont les suivants :

- nom de la machine virtuelle : myvm
- adresse du réseau LAN partagé par le domm0 et le domU : 192.168.1.0
- adresse du pont xenbr0 dans le réseau LAN partagé par le dom0 et le domU :192.168.1.101
- adresse de l'interface eth0 du domU : 192.168.1.102

2.2.1 Configuration au niveau du dom0

Nous avons choisi le mode bridge (script = vif-bridge). Dans le fichier de configuration de la machine virtuelle (/etc/xen/myvm.cfg), au niveau du paramètre vif, ajouter les paramètres script et bridge comme dans l'exemple suivant : vif = [ip=10.0.0.1, mac=00 :16 :3E :8B :54 :20, script=vif-bridge,bridge=xenbr0']

Ensuite il faut définir le pont utilisé par le domU ceci se fait au niveau du paramètre vif du fichier /etc/xen/myvm.cfg (bridge=xenbr0).

Créer le bridge qui sera utilisé pour la connexion du domU au dom0 avec la commande `brctl addbr xenbr0`

Attribuer une adresse IP au bridge xenbr0 dans le LAN partagé par le dom0 et le domU avec la commande `ifconfig xenbr0 192.168.1.101/24`

activer le bridge xenbr0 avec la commande `ifconfig xenbr0 up`

2.2.2 Configuration au niveau du domU

Après avoir démarré la VM, exécuter les commandes suivantes :

- Consulter les interfaces présentes avec la commande `ip a`
- Activer l'interface eth0 avec la commande `ip link set eth0 up`
- Attribuer une adresse ip à l'interface eth0 dans le réseau lan partagé par le dom0 et le domU avec la commande `ip addr add 192.168.1.102/24 dev eth0`
- définir l'interface eth0 comme passerelle vers le réseau LAN partagé par le dom0 et le domU avec la commande `ip route add 192.168.1.0/24 dev eth0`

2.3 Connexion au domU à internet

Pour pouvoir configurer l'environnement pour l'exécution de l'application que nous avons choisie sur notre machine virtuelle, et également mettre en place une connexion ssh avec le dom0, il sera nécessaire que notre machine virtuelle ait accès à internet pour permettre le téléchargement et l'installation des paquets. Pour cela, nous devons mettre en place un NAT (Network Address Translation) au niveau du dom0 pour transmettre tous les paquets destinés au réseau extérieur provenant du LAN interne partagé par le dom0 et le domU.

2.3.1 Configuration au niveau du dom0

voici les étapes à suivre :

- Vérifier l'interface ethernet nous donnant accès à internet avec la commande `ifconfig`
- configurer les tables des adresses ip avec NAT
- Activer le forwarding des paquets au niveau du noyau du système avec la commande `echo 1 > /proc/sys/net/ipv4/ipforward`
- Installer le paquet `iptables-persistent` pour la persistance des configurations effectuées sur `iptables`.

2.3.2 Configuration au niveau du domU

voici les étapes à suivre :

- définir la passerelle par défaut pour les paquets destinés à internet via la commande `ip route add default via 192.168.1.101 dev eth0`
- Indiquer l'adresse du serveur DNS en allant dans le fichier `/etc/resolv.conf` et le modifier comme suit : `nameserver 8.8.8.8`
- Activer le forwarding des paquets au niveau du noyau du système avec la commande `echo 1 > /proc/sys/net/ipv4/ipforward`
- Tester la connexion à internet avec un ping

Chapitre 3

IMPLEMENTATION

3.1 Architecture de la solution

Nous avons decide de diviser notre solution en 03 modules :

- Un module permettant d'automatiser la creation et la configuration de la machine virtuelle
- un module permettant la selection des features essentiels au fonctionnement de l'application et la machine virtuelle
- un module permettant la creation et le demarrage d'une nouvelle machine virtuelle avec les features utiles dans le fichier de configuration de xen.

3.1.1 Module de configuration et de demarrage de la machine virtuelle

Il s'agit ici de developper un script bash permettant d'executer toute les commandes de lancement et de configuration de la machine virtuelle. ce script contiendra aussi les commandes de configuration du domU et dom0.

3.1.2 Module d'extraction des features essentiels au fonctionnement de la machine virtuelle et de l'application

Pour ce module le procede utilise est le suivant :

- ecrire un programme c capable d'extraire les features de la machine virtuelle
- ensuite ecrire un script qui desactive chaque feature dans le fichier de configuration de la machine virtuelle
- relancer la machine virtuelle et l'application pour savoir si le feature desactive est important ou pas.

3.1.3 Module de configuration de la machine virtuelle pour la migration

une fois les deux premiers modules fait il s'agira ici d'ecrire un script bash capable :

- modifier le fichier de configuration de l'hyperviseur en lui donnant les valeurs de la liste de features essentiels aux fonctionnement de la machine virtuelle et l'application
- creer et demarrer une nouvelle machine virtuelle en utilisant le fichier de configuration modifie