



Xen : Principes et Mise en oeuvre

D.Caron

¹LPTA UMR 5207, CNRS, Université Montpellier 2, France

SUPAgro INRA Montpellier - 3,4 et 5 Decembre 2007



<http://laser.cbs.cnrs.fr/>

Sommaire

1 Introduction

- Pourquoi virtualiser ?
- Les Rings
- Principes de la Virtualisation
- La virtualisation selon Xen

2 Architecture de Xen

3 Le réseau dans Xen

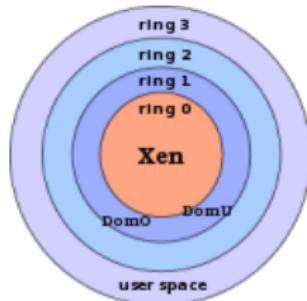
4 Gestion des domaines DomU

5 Installation de Xen

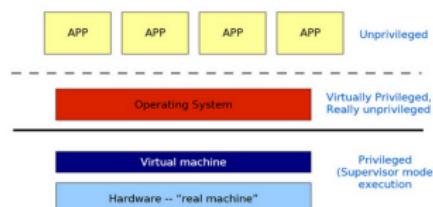
6 Conclusions et Références

- 2007 marque la disparition des machines monoprocesseurs.
- La course aux Megahertz est (enfin) arrêtée au profit de la performance/watt.
- Les consommations restent élevée et les machines surperformantes pour de nombreuses utilisations.
- L'utilisation florissante de la virtualisation est née de ces constats:
On ne peut pas mettre tous les services sur une même machine, les machines sont devenues trop puissantes pour ce que l'on en fait, comme elles consomment beaucoup d'énergie, on gaspille donc énormément et ce n'est plus le moment.
- La virtualisation permet de mieux exploiter les performances et ainsi baisser la consommation d'énergie à service équivalent.
- Installer un serveur Intranet et un serveur Extranet sur la même machine est devenu possible grâce à la virtualisation.

Toutes les instructions d'un processeur (ISA) ne sont pas sur un pied d'égalité. Parmi elles, certaines peuvent modifier la configuration des ressources du processeur ; elles sont dites critiques. Pour protéger la stabilité de la machine, ces instructions ne peuvent pas être exécutées par tous les logiciels. Du point de vue du CPU, les logiciels appartiennent à 4 catégories, ou niveaux d'abstraction : les anneaux (rings) 0, 1, 2, 3. Chaque anneau définit un niveau de privilège décroissant. Les instructions les plus critiques réclament les priviléges les plus élevés, d'ordre 0.



Staying in Control

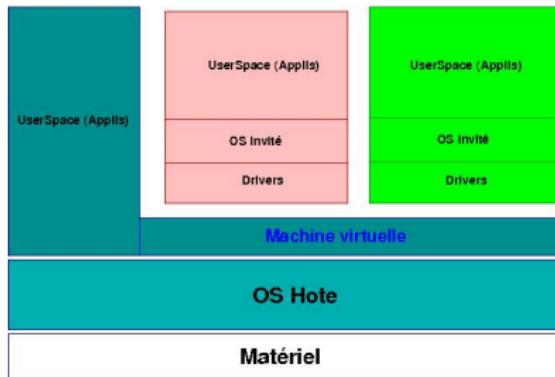


C'est

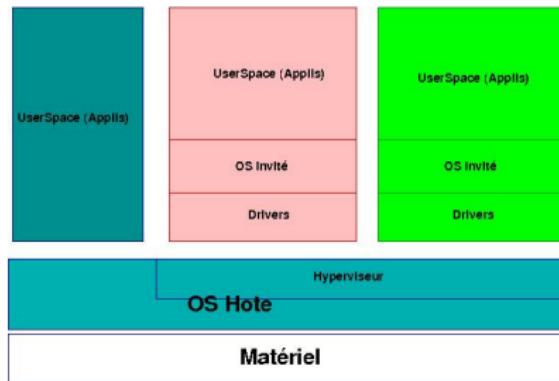
pourquoi plusieurs OS (ring 0) ne peuvent tourner sur la même machine physique.

- Couche d'abstraction matérielle et/ou logicielle.
- On appelle OS hôte le système directement installé sur le matériel.
- On appelle OS invité(s) le ou les OS installés sur le système hôte.
- Un système virtuel partitionne, partage et/ou isole des ressources physiques ou logicielles.
- Les OS invités sont manipulables par le l'OS hôte
démarrage, arrêt, gel, clonage, migration etc...
- l'OS hôte possède un 'réseau virtuel', purement logiciel entre hôte et invités.

- Il existe 3 grands types de virtualisations.
- Les 'isolants' (chroot,Linux-VServer...) ne font qu'isoler certains aspects ou ressources. Très performants mais les OS invités ne sont que (très) peu isolés.
- La virtualisation 'simple' (vmware, virtualbox...) consiste à émuler du hardware (avec son BIOS). Très gourmande en ressource, elle a l'avantage d'être universelle.



- La paravirtualisation implique que le noyau de l'OS hôte soit modifié et adapté pour faire tourner des OS invités. Le noyau modifié contient l'hyperviseur' ou VMM (Virtual Machine Monitor). L'hyperviseur se charge de superviser les accès aux instructions du ring 0.
- La paravirtualisation permet de se rapprocher de performances proches de celles obtenues nativement par un système d'exploitation.
- Xen 3 peut également faire tourner des systèmes non modifiés sur des processeurs supportant la VT.



Sommaire

1 Introduction

2 Architecture de Xen

- L'hyperviseur
- Dom0, le domaine privilégié
- DomU, domaine non privilégié
- Les périphériques
- Le démon Xend

3 Le réseau dans Xen

4 Gestion des domaines DomU

5 Installation de Xen

6 Conclusions et Références

L'architecture d'un système Xen est composée.

- D'un Hyperviseur (dans le noyau modifié).
 - Il gère les temps d'utilisation CPU de chaque domaine. Dans leur temps alloué par le superviseur les OS invités ordonnancent leurs processus.
 - Déetecte et démarre les processeurs non initialisés par le BIOS.
 - Supervise les interruptions (instructions privilégiées).
- De machines virtuelles appelées domaine.
 - Dom0 - c'est le domaine privilégié, c'est dans ce domaine que tournent les applications permettant de contrôler les autres machines virtuelles. On peut y faire tourner des services comme dans tout autre domaine bien que cela ne soit pas recommandé pour des raisons de sécurité.
 - DomU - Toutes les autres machines virtuelles sont dans des domaines non privilégiés appelés DomU. Ces domaines sont sous le contrôle du Dom0.

Dom0, le domaine privilégié.

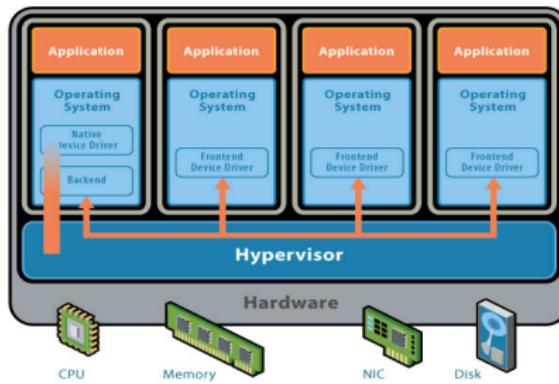
- Il est lancé au boot de la machine physique.
- C'est dans ce domaine que tourne le noyau LinuX modifié et les applications de contrôle des DomU.
- Seul l'hyperviseur tourne dans le ring 0, le Dom0 tourne dans le ring 1 et reste sous le contrôle de l'hyperviseur.
- Il agit en direct avec le matériel via les pilotes du noyau.
- Le démon xend , chargé des tâches d'administration, tourne dans ce domaine.
- Assure la liaison pilotes natifs <=> pilotes virtuels des domaines.
- Contient les fichiers de configuration des DomU (noyau à démarrer, disques, mémoire, réseau....)

DomU, les domaines non privilégiés.

- Plusieurs DomU peuvent être lancés et contrôlés par le Dom0. Dans ces domaines tournent les hôtes invités.
- Leur noyau est chargé par le Dom0 et tourne dans un ring non privilégié (ring 1 ou 2).
- Les noyaux chargent des pilotes virtuels ou natifs selon que le processeur supporte ou pas la VT.

Gestion des périphériques (drivers)

- Le Dom0 possède les pilotes natifs nécessaires pour l'accès aux périphériques.
- Le Dom0 présente un backend de communication aux pilotes virtuels des DomU.
- Les DomU possèdent des pilotes virtuels et accèdent aux périphériques réels via le backend.



Le démon Xend

- Xend, écrit en python, tourne dans le Dom0 et répond aux requêtes des applications de gestion Xen (commandes xm) issues du Dom0.
- Il peut présenter une interface Web de gestion sur le port 8000.
- Fichier de configuration : */etc/xen/xend-config.sxp*
 - Configuration du réseau.
 - Gestion de la mémoire minimum pour Dom0.
 - CPU(s) de Dom0.
 - Paramètres de migration.
- Lance deux démons: xenstored (contrôle des ressources allouées selon *xend-config.sxp*) et xenconsoled (contrôle les connexions aux consoles des DomU - commande 'xm console').
- Fichiers de log : */var/log/xend.log* et */var/log/xend-debug.log*



Sommaire

1 Introduction

2 Architecture de Xen

3 **Le réseau dans Xen**

- Les Interfaces virtuelles
- Les Modes réseau: Le mode Bridge
- Les Modes réseau: Le mode NAT
- Les Modes réseau: Le mode Route

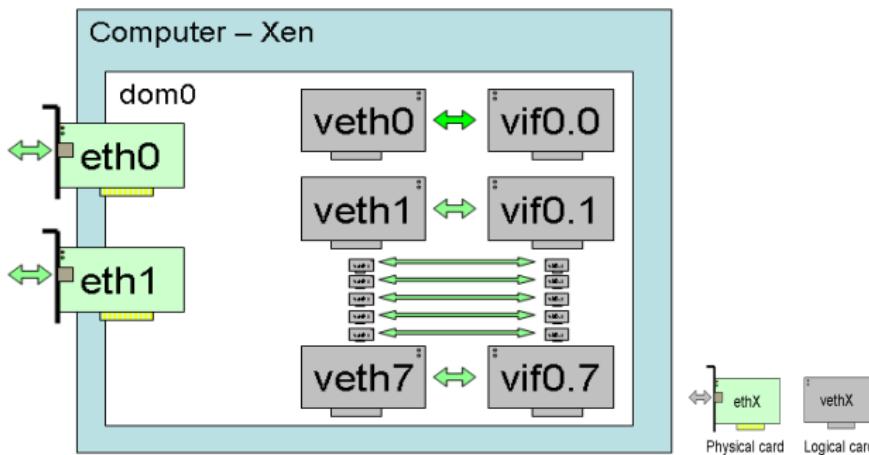
4 Gestion des domaines DomU

5 Installation de Xen

6 Conclusions et Références

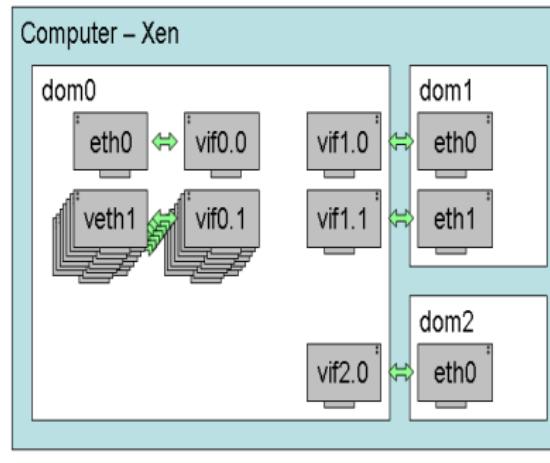
Interfaces virtuelles (vif)

- Xen crée 7 paires d'interfaces virtuelles ethernet utilisable par le Dom0.
- Ces paires vethx <=> vif0.x sont reliées par une cable croisé.



Interfaces virtuelles (vif)

- Pour chaque DomU (numéroté #N), Xen crée des paires d'interfaces virtuelles ($vifN.x <=> ethx$)
- Les $vifN.x$ sont dans le Dom0. Dans les DomU le nom des interfaces est $ethx$ (sous LinuX).

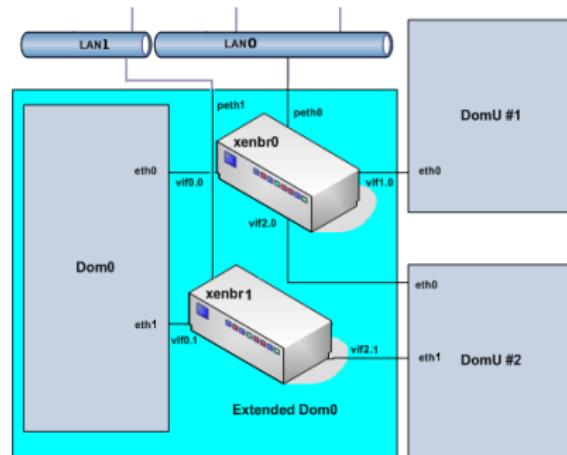


Le Mode Bridge

- C'est le mode par défaut. Xen utilise un ou des ponts (bridges) à l'intérieur du Dom0. Ce mode permet à tous les DomU d'apparaître comme des machines individuelles.
- Au démarrage xend exécute le script network-bridge qui:
 - Crée un ou des ponts (xenbrx)
 - L'interface physique ethx est arrêtée. L'adresse MAC et l'IP de ethx sont 'recopiés' sur l'interface virtuelle vethx.
 - L'interface physique ethx est renommée pethx. L'interface virtuelle vethx est renommée ethx.
 - **pethx et vif0.x** sont attachés au pont xenbrx.
 - Le pont **xenbrx**, **pethx** et **vif0.x** sont 'démarrés'.

Le Mode Bridge (2)

- **pethx** : interface physique (configurée par Xen)
- **ethx** : interface virtualisée (ex **vethx**) configurable (configuration de **ethx** physique).
- **vifN.x** interfaces virtuels du ou des ponts Xen vers le DomU #N.



Le Mode Bridge (3)

```
# /bin/ifconfig
eth0 Link encap:Ethernet HWaddr 00:13:72:5D:FA:C0
inet addr:162.38.125.4 Bcast:162.38.125.255 Mask:255.255.255.0
...
eth1 Link encap:Ethernet HWaddr 00:13:72:5D:FA:C1
inet addr:162.38.243.53 Bcast:162.38.243.255 Mask:255.255.255.0
...
peth0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::fcff:ffff:feff:ffff/64 Scope:Link
...
peth1 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::fcff:ffff:feff:ffff/64 Scope:Link
...
vif0.0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::fcff:ffff:feff:ffff/64 Scope:Link
...
vif0.1 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::fcff:ffff:feff:ffff/64 Scope:
...
vif1.0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::fcff:ffff:feff:ffff/64 Scope:Link
...
vif2.0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::fcff:ffff:feff:ffff/64 Scope:Link
...
vif2.1 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::fcff:ffff:feff:ffff/64 Scope:Link
...
xenbr0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::200:ff:fe00:0/64 Scope:Link
...
xenbr1 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
```

Le Mode Bridge (4)

- Les paquets sortants des DomU via vifN.x ne peuvent pas être contrôlés par le Dom0 (iptables) (voir figure).
- Les vifN.x n'ont pas d'adresses IP.
- Pour utiliser le mode Bridge, il faut installer le paquetage bridge-utils.

```
# rpm -aq | grep bridge
bridge-utils-1.0.4-4
```

Le Mode NAT

- Dom0 joue le rôle de passerelle pour les DomU.
- les vifN.x ont pour IP celles des cartes des DomU.
- Les règles iptables sont applicables à ces cartes dans le Dom0.
- Pour activer le mode NAT il faut modifier le fichier de configuration de xend:

```
#(network-script network-bridge)
#(vif-script vif-bridge)
(network-script network-nat)
(vif-script vif-nat)
```

Le Mode NAT (2)

- Configurer également le(s) DomU pour utilisation du NAT dans les fichiers de config Xen des DomU.
- ### Example:

```
vif=[ 'ip=10.0.0.2' ]
dhcp="off"
ip="10.0.0.2"
netmask="255.255.0.0"
gateway="10.0.0.254"
hostname="monDomU"
```
- Les numéros IP DOIVENT être dans le réseau 10.x.x.x (c'est ainsi que Xen est configuré).

PS: Ce Mode ne semble pas encore fonctionner correctement sur Xen?

Le Mode Route

- Le Dom0 devient routeur pour les DomU.
- Les vifN.x ont pour IP celles des cartes des DomU.
- Elles ne voient pas passer les paquets.
- Les règles iptables sont applicables à ces cartes dans le Dom0.
- Pour activer le mode Route il faut modifier le fichier de configuration de xend:

```
#(network-script network-bridge)
#(vif-script vif-bridge)
(network-script network-route)
(vif-script vif-route)\
```

Sommaire

1 Introduction

2 Architecture de Xen

3 Le réseau dans Xen

4 Gestion des domaines DomU

- Arrêt/Démarrage
- Gestion de la mémoire
- Gestion de la CPU
- Gestion des états
- Migration de domaine

5 Installation de Xen

6 Conclusions et Références

Arrêt/Démarrage des domaines DomU

- Pour qu'un domaine démarre automatiquement au boot de la machine physique il faut et il suffit que son fichier de configuration soit dans le répertoire:`/etc/xen/auto`
- La commande `xm [command]` (Xen Management) est la commande de gestion principale.
- `# xm list`
Affiche la liste de tous les domaines en fonctionnement.
- `# xm create [-c] <ConfigFile>`
Démarre le DomU défini dans le fichier de configuration `<ConfigFile>`, l'option -c attache une console au DomU.
- `# xm shutdown [-a|-w] <DomainName>`
Arrête le domaine `<DomainName>`, l'option -a (all) arrête tous les domaines, -w (wait) ne rend la main qu'une fois le domaine complètement arrêté.
- `# xm console <DomainName>`
Attache une console au domaine `<DomainName>`

Attribution/Gestion de la mémoire des domaines

- C'est l'hyperviseur qui contrôle les allocations de mémoire.
- On n'attribut pas de mémoire au Dom0, il prend ce qui reste de mémoire physique.
- C'est dans le fichier de configuration du domaine DomU que l'on spécifie la taille mémoire qui lui est attribuée.
- On peut modifier la quantité de mémoire allouée à un DomU pendant son fonctionnement, sans jamais pouvoir passer sous la taille fixée dans son fichier de configuration.
- `# xm mem-set <DomainName> <Mem>`
Modifie la quantité de mémoire (en Mo) allouée au domaine `<DomainName>`.

Attribution/Gestion de la mémoire des domaines

- C'est l'hyperviseur qui contrôle les allocations CPU.
- VCPU = Virtual Central Process Unit. Pour Xen 1 proc = 1 CPU.
Dans un système Hyperthread, 1 thread = 1 VCPU.
- C'est dans le fichier de configuration du domaine DomU que l'on spécifie le ou les VCPUs qu'il peut utiliser.
- On peut modifier le nombre de VCPU alloués à un DomU pendant son fonctionnement.
- `# xm vcpu-set <DomainName> <nVCPU>`
Affecte au domaine <DomainName> <nVCPU> VCPU(s).
- `# xm vcpu-list`
Liste les VCPUs de la machine, leur affectation aux DomU et les temps d'utilisation par les DomU.
- `# xm vcpu-pin <VCPU> <CPUs>`
Force un DomU à utiliser un VCPU donné.

Gestion des états

- Il existe 3 états pour les DomU.
- L'état 'Running', c'est l'état normal d'un DomU.
- L'état 'Pause', le DomU est en pause, ses ressources lui restent affectées, il apparaît dans la liste des DomU. Il ne peut faire aucun accès à ses ressources. Au niveau de l'hyperviseur, il est gelé.
- `# xm [pause|unpause] <DomainName>`
Met ,ou sort le <DomainName> de l'état de pause.
- L'état 'Hibernation', le DomU s'arrête, les ressources sont restituées au Dom0, mais son état système est enregistré dans un fichier.
- `# xm save <DomainName> <CheckpointFile>`
Met le <DomainName> en état d'hibernation et sauve son état système dans le fichier <CheckpointFile>.
- `# xm restore <CheckpointFile>`
Reprise de l'exécution du DomU dont l'état système est enregistré dans le fichier <CheckpointFile>.

Migration de domaine

- La Migration est le transfert d'un domaine entre 2 machines physiques différentes.
- Les 2 machines DOIVENT avoir accès à l'image disque où est installé le domaine (File System en réseau impliquant un réseau rapide entre le serveur de FS et les DomU), la version de Xen DOIT être la même sur les deux machines.
- **'Regular' Migration:**

```
# xm migrate <DomainName> <XenHost>
```

Migration du domaine <DomainName> vers la machine Xen <XenHost>. Lors d'une migration 'regular', le domaine est mis en pause.

- **'Live' Migration:**

```
# xm migrate --live <DomainName> <XenHost>
```

Le DomU n'est PAS mis en pause. La durée de l'arrêt effectif est de l'ordre de la centaine de millisecondes.

- Aucune connexion réseau n'est perdue lors d'un migration.



Sommaire

- 1 Introduction
- 2 Architecture de Xen
- 3 Le réseau dans Xen
- 4 Gestion des domaines DomU
- 5 Installation de Xen
 - Partitionnement de l'espace disque
 - Installation du Dom0
 - Configuration du Dom0
 - Configuration des DomU
 - Installation d'un DomU
- 6 Conclusions et Références

Partitionnement de l'espace disque

- Le partitionnement de l'espace disque dépend de l'utilisation du serveur Xen (nombre de DomU, services des DomU etc...)
- Comme pour tous serveurs mais encore plus dans le cas de serveurs Xen, les partitions de type LVM sont recommandées.
- Le Dom0 devant être le plus sécurisé possible, il est recommandé de faire tourner le minimum de service sur le Dom0 ce qui implique un espace disque réduit au strict minimum.
- Tous les DomU se partageront le reste du disque (si ils ne sont installés sur un serveur de fichier en réseau permettant la migration).
- Les répertoires `/etc` /`var` /`tmp` etc... des DomU devant être par nature différents (configuration, log, répertoire web, mail etc...) il n'en est pas de même pour `/usr`, `/usr/local` etc... qui pourront donc être écrits sur un (des) filesystem(s) partagé(s) par les DomU.
- Ne pas oublier les espaces de swap des DomU.

Installation du Dom0

- Vérifier que le paquetage bridge-utils est bien installé sur la machine.
- Télécharger les paquetages Xen-3.1 depuis
<http://xen.xensource.com/xen/downloads> (choisir les paquetages RHEL4.5 : xen-3.1.0-1.i386.rpm, kernel-xen-2.6.18-3.1.0.i386.rpm).
- Télécharger également les paquetages RPM de la glibc 'xénifiée' sur
<http://xenbits.xensource.com/glibcs/rhel44/RPMS/i386/>.
- Installer tous ces paquetages. # *rpm -ivh *.rpm* .
- Editer le fichier */etc/grub.conf*.

```
title Xen 3.1.0
root (hd0,0)
kernel /xen-3.gz dom0_mem=262144
module /vmlinuz-2.6-xen0 root=/dev/VolGroup00/LogVol00 ro
maxcpus=1 console=tty0 nousb noreboot \
module /initrd-2.6-xen.img
```

Configuration du Dom0

- Editer le fichier `/etc/xen/xend-config.sxp`

- Vérifier la configuration réseau (Brigde, NAT ou Route):
`(network-script network-xxx)`
`(vif-script vif-xxx)`

Remplacer xxx par bridge,nat ou route. Pour des configurations avancées de réseau, créer ses propres scripts.

- Attribuer le(s) CPU(s) et la mémoire minimum du Dom0.

`(dom0-min-mem 192)`
`(dom0-cpus 1)`

- Configurer la Migration.

```
# Cette machine accepte (yes) ou non (no) les migrations
(xend-relocation-server yes)

# Le port de connexion pour la migration
(xend-relocation-port 8002)

# Les machines autorisées à migrer leur DomU
(xend-relocation-hosts-allow '^localhost$')
```

Configuration des DomU

- La configuration Xen des DomU se fait dans des fichiers se trouvant sur le Dom0. Il faut un fichier de configuration par DomU.
- Les fichiers de configuration des DomU se trouvent n'importe où sur le Dom0!

Sauf si l'on veut que les DomU démarrent au boot de la machine physique auquel cas ces fichiers doivent être dans `/etc/xen/auto`.

- **Attention**, Tous comme les fichiers de configuration d'un système respectent la configuration hardware d'une machine, les fichiers de configuration système des DomU doivent tenir compte de la configuration Xen du domaine.

Par exemple , si Xen a attribué 3 filesystems à un DomU, le fichier `/etc/fstab` de ce DomU devra respecter cette configuration.



Fichier de configuration d'un DomU

Les paramètres les plus importants:

```
# Kernel image file
kernel = "/boot/vmlinuz-2.6.16-xen"
# Optional ramdisk
ramdisk = "/boot/initrd-2.6.16-xen.img"
# Initial memory allocation (in megabytes) for the new domain.
memory = 316
# A name for your domain. All domains must have different names
name = "LPTA_Intranet"
# Create a Vnic with sensible defaults using an empty vif clause
vif = [ '' ]
# Define the disk devices you want the domain to have access to
disk = ['phy:VolGroup001/lvol5,hda1,w','phy:VolGroup001/lvol4,hda2,w']
# Set root device.
root = "/dev/hda1 ro"
# Sets runlevel 3
extra = "3"
```

Installation d'un DomU

- Les installateurs LinuX modernes sont capables de détecter les partitions LVM sur un disque. Pour installer le système d'un DomU on reboote donc la machine sur un CD d'install et on indique à l'installateur sur quel LVM on désire installer le système. Ne pas installer de bootloader.
- Rebooter le Dom0, télécharger les paquetages Xen-3.1 depuis xen.xensource.com/xen/downloads (choisir les paquetages RHEL4.4 : kernel-xenU-2.6.9-42.0.3.EL.xs0.4.0.263.i686.rpm), monter la ou les partitions contenant le DomU et faire un chroot dans la partition /root du DomU.
- Modifier les fichiers */etc/fstab*, */etc/modprobe.conf*, */etc/sysconfig/network-script/ifcfg-ethn*.
- Installer le paquetage du noyau modifié de domU (*kernel-xenU*) et les paquetages de la glibc xénifiée préalablement téléchargés.

Installation d'un DomU (2)

- Sortir du chroot de la partition du DomU et tester la configuration du DomU.

```
# xm -c create monDomU.cfg
```
- La console de boot du DomU *monDomU* va apparaître, suivre le déroulement du boot.
- Si le boot s'est bien déroulé, on peut alors se loguer sur le DomU et vérifier que le système est fonctionnel.
- Dans Dom0, déplacer le fichier *monDomU.cfg* dans */etc/xen/auto* pour que le domaine démarre au prochain reboot de la machine.
- **On ne trouve aucun fichier spécifique à Xen sur le(s) filesystem(s) d'un DomU.**

Sommaire

- 1 Introduction
- 2 Architecture de Xen
- 3 Le réseau dans Xen
- 4 Gestion des domaines DomU
- 5 Installation de Xen
- 6 Conclusions et Références
 - Conclusions
 - Références
 - Travail Dirigé

Conclusions

- Finalement, on constate que l'installation d'un système Xen est relativement facile.
- La partie la plus délicate est probablement la configuration du réseau interne de Xen si l'on veut une architecture réseau un peu exotique.
- La maintenance de Xen est quasi nulle. Chaque DomU se gère comme une machine physique. Xen disparaît et ne se manifeste jamais, sauf si on fait appelle à lui (par exemple pour une migration ou que l'on constate que l'on a été un peu 'léger' sur la mémoire de tel DomU) mais c'est alors toujours pour faciliter la vie de l'Administrateur.
- Si l'on a pris soin d'installer les DomU sur des partitions LVM, la sauvegarde d'un DomU se fera depuis le Dom0 en faisant un snapshot (`lvcreate -s`) de la partition du DomU puis en sauvegardant ce snapshot.

Références

- **Laser:** Diaporams sur la virtualisation.
- **RAISIN:** TP Xen.
- **DELL:** Live Migration with Xen Virtualization Sofware.
- **XenWiki**
- **Mitopia Wiki:** Xen Setup.

Merci de Votre Attention et Bon TD !!



Des Questions? ... avant de commencer!

Travail Dirigé

- Installer un Dom0 sur votre machine (il acceptera les migrations).
- Partitionner l'espace disque vierge (en LVM pour ceux qui veulent 3 points de plus!) pour y faire 4 FileSystems (Fs1, Fs2, Fs3 et Fs4) . Fs1 sera formatté en swap (1Go), les 3 autres en ext3.
- Fs2 et Fs3 contiendront les racines / respectives des DomU *LOCDom* et *NFSDom*, Fs4 contiendra /usr de ces DomU.
- Configurer 2 DomU (*LOCDom* et *NFSDom*):
 - *LOCDom* aura son swap, sa racine et /usr sur les Filesystems Fs1, Fs2 et Fs4.
 - *NFSDom* sera un système monté en NFS (sur les répertoires /NFSDom/root et /NFSDom/usr partagés du Dom0).

```
root = "/dev/nfs"
nfs_server = '192.168.X.Y'
nfs_root = '/NFSDom/root'
```

Travail Dirigé (2)

- Installer un système DomU sur les FileSystems Fs1, Fs2 et Fs4 (partitions *swap*, */* et */usr*).
- Recopier 'à la main' la partition contenant la racine (*/* sur Fs2) du DomU *LOCDom* précédemment installé sur la partition Fs3), modifier les fichiers de configuration système (*/etc/fstab*, */etc/sysconfig/network/ifcfg-eth0*, */etc/hostname*)
- La partition Fs3 sera montée au boot du Dom0 sur */NFSDom/root* et exportée en NFS rw.
- La partition Fs4 sera montée au boot du Dom0 sur */NFSDom/usr* et exportée en NFS ro.
- Le DomU NFSDom aura sa partition racine montée en NFS (pour pouvoir faire une migration):