



Alphorm.com

Présentation de la formation

La virtualisation légère LXC

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Présentation du formateur
- Le plan de formation
- Publics concernés
- Connaissances requises
- Liens utiles



Formation Virtualisation légère avec LXC

alphorm.com™©

Présentation du formateur

- Ludovic QUENEC'H DU
- lquenec@alphorm.com
- Consultant et expert en Open source, logiciel libre et virtualisation
- Mission de conseil, d'architecture, d'administration, de migration et de formation
- Mes références :
 - Mon profil [LinkedIn](https://fr.linkedin.com/pub/ludovic-quenec-hdu/47/6bb/550) : <https://fr.linkedin.com/pub/ludovic-quenec-hdu/47/6bb/550>
 - Mon profil [Alphorm](http://www.alphorm.com/formateur/ludovic-quenechdu) : <http://www.alphorm.com/formateur/ludovic-quenechdu>



Mes formations sur Alphorm

Les formations de Ludovic QUENECHDU

 OpenVZ QUENECHDU Ludovic 30/07/2015 5h57m ★★★★★	 VirtualBox QUENECHDU Ludovic 02/07/2015 6h21m ★★★★★	 VMware Workstati... QUENECHDU Ludovic 04/06/2015 3h51m ★★★★★	 KVM QUENECHDU Ludovic 07/05/2015 7h39m ★★★★★
 Linux LPIC-3 Enviro... QUENECHDU Ludovic 30/01/2015 20h14m ★★★★★	 XenSources QUENECHDU Ludovic 02/07/2014 12h10m ★★★★★	 ProxmoxVE 3.x QUENECHDU Ludovic 26/02/2014 7h1m ★★★★★	

Le plan de formation

- Concept
 - La virtualisation par conteneurs
 - Les Namespaces et les Control groups
 - LinuxContainers.org – LXC
- Mise en œuvre
 - Installation et configuration de LXC
 - Containers privilégiés
 - Containers non-privilégiés
- Exploitation des containers avec LXC
 - Commandes de manipulation des conteneurs
 - Le clonage des conteneurs
 - Snapshot
 - Gestion des ressources avec les cgroup
- Exploitation des containers avec LXD
 - LXD présentation et installation
 - Gestion des images LXD
 - Manipulation des conteneurs
 - Les snapshots
- Exploitation des containers avec LXD
 - Administration des serveurs distants
 - Migration des conteneurs
- Configuration avancée des containers
 - Le modèle réseau Veth, MacVlan, le Bridge et Phys
 - OpenVswitch et LXC
 - Les Backends de stockage
 - LVM et LXC
 - ZFS et LXC
 - BTRFS et LXC

Publics concernés

- ✓ Chef de projet
- ✓ Administrateur
- ✓ Développeur
- ✓ Les techniciens

Connaissances requises

- ✓ Un bonne culture du système Linux
- ✓ La maitrise de la ligne de commande shell
- ✓ Des connaissances sur le stockage et le réseau
- ✓ Une connaissance de la virtualisation

Liens utiles

- <https://linuxcontainers.org/fr/>
- <https://github.com/lxc/lxd>
- <https://github.com/lxc>
- <https://www.stgraber.org/category/lxc/>

Autres formations sur la virtualisation

Thèmes :

Toutes (81)

Virtualisation (17)

Systèmes (12)

Programmation (21)

Base de données (8)

OpenSource (3)

Messagerie (2)

Réseaux (1)

Sécurité (1)

Scientifique (2)

Scripting (2)

Utilisateur (8)

Marketing (2)

Collaboration (2)

Les plus récentes :

Windows 10

Windev 20. Perfectionne...

OpenVZ

KnockoutJS

VirtualBox

JavaFX

Citrix XenDesktop 7.5 Get...

Virtualisation

17 formations

Trier par: Les + récentes

Première Précédent 1 2 Suivant Dernier



OpenVZ

Ludovic QUENECHDU

30/07/2015 00:00:00

5h57min 24 Vidéos

★★★★☆



VirtualBox

Ludovic QUENECHDU

02/07/2015 00:00:00

6h21min 30 Vidéos

★★★★☆



Citrix XenDesktop...

FABRICE SFORZA CHR...

18/06/2015 00:00:00

8h49min 39 Vidéos

★★★★☆



VMware Workstat...

Ludovic QUENECHDU

04/06/2015 00:00:00

3h51min 21 Vidéos

★★★★☆



KVM

Ludovic QUENECHDU

07/05/2015 00:00:00

7h39min 28 Vidéos

★★★★☆



XenSources

Ludovic QUENECHDU

02/07/2014 13:30:00

12h10min 34 Vidéos

★★★★☆



ProxmoxVE 3.x

Ludovic QUENECHDU

26/02/2014 19:42:00

7h01min 30 Vidéos

★★★★☆



Citrix XenServer 6 ...

FABRICE SFORZA CHR...

23/06/2013 10:59:00

9h10min 33 Vidéos

★★★★☆



Veeam Backup & ...

Stéphane HEBERLE

11/06/2013 18:35:00

5h42min 18 Vidéos

★★★★☆

Formation Virtualisation legere avec LXC

Laissez-nous un commentaire

alphorm.com™©

Are you ready ? ☺





Alphorm.com

Concept

La virtualisation par conteneurs

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Le modèle : Machines Virtuelles
- Le modèle : isolation, containers
- Les produits : "Isolateur", containers

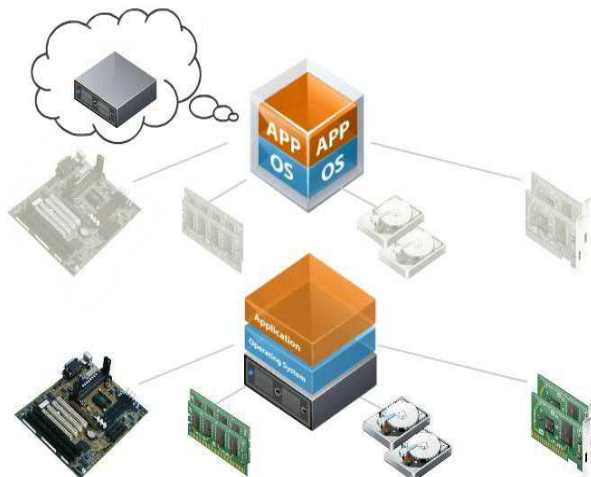


Formation Virtualisation légère avec LXC

alphorm.com™©

Le modèle : Machine Virtuelle

- Une machine virtuelle c'est :
 - Représentation logicielle d'une machine physique
 - Émulation du CPU, mémoire, nic, contrôleur de disque, disque, etc..
 - Un OS installé dans la VM – Guest OS ou OS invité



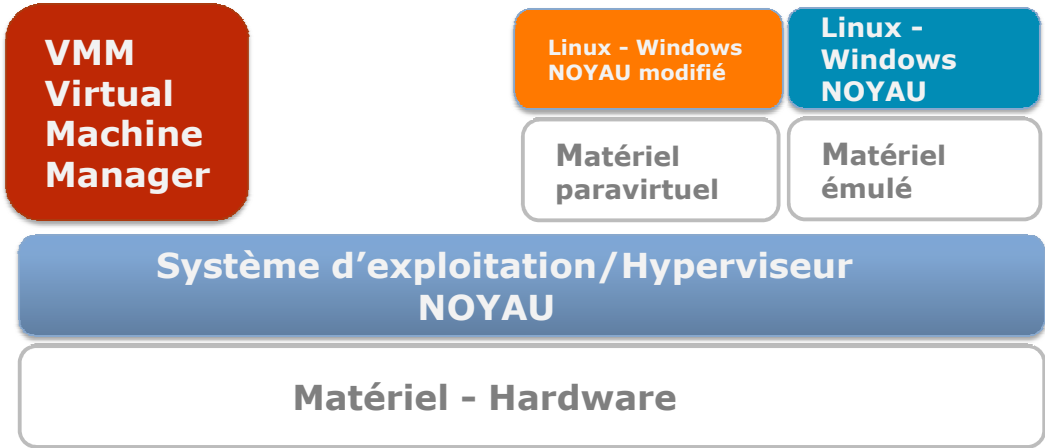
Le modèle : Machine Virtuelle

- La virtualisation complète



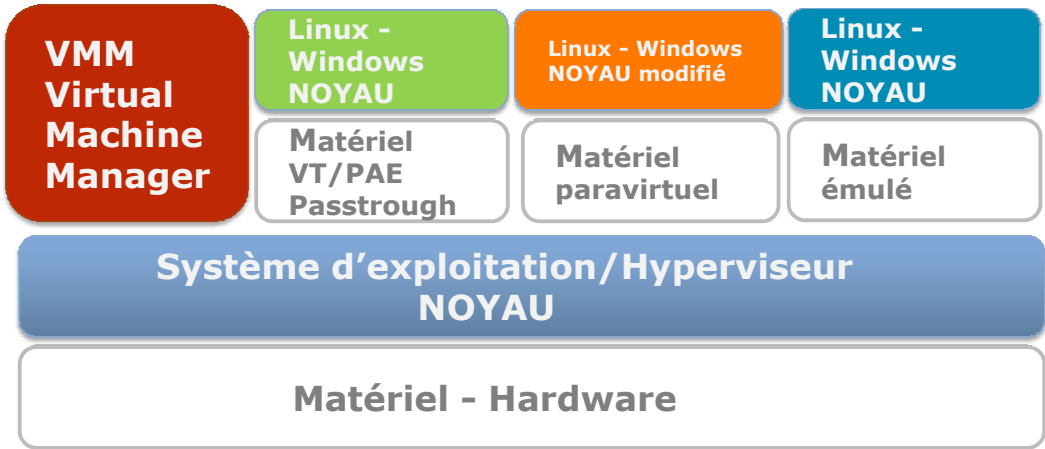
 Le modèle : Machine Virtuelle

- La virtualisation complète
- La para virtualisation



 Le modèle : Machine Virtuelle

- La virtualisation complète
- La para virtualisation, La virtualisation assistée par le matériel





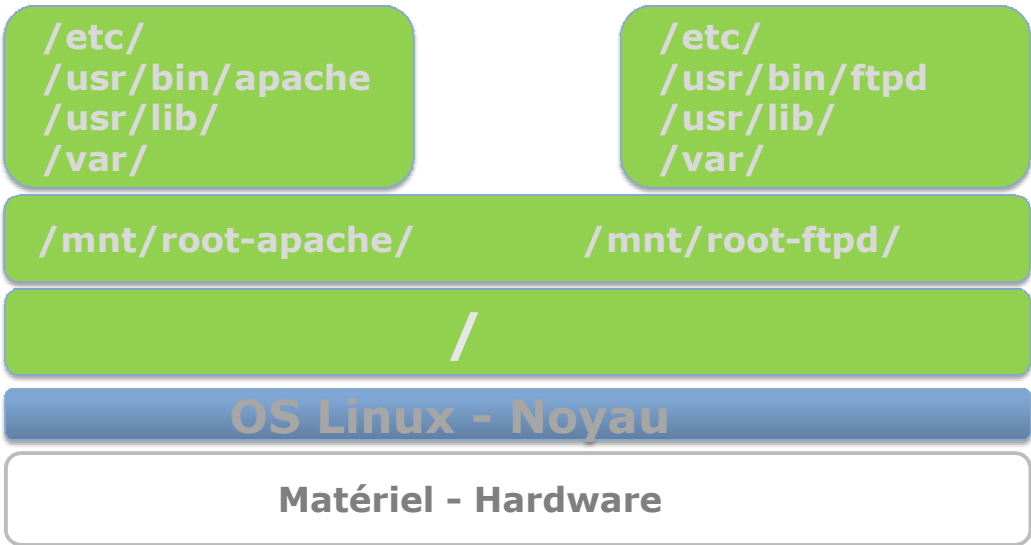
Le modèle : isolation, containers

- Mode de fonctionnement « classique »



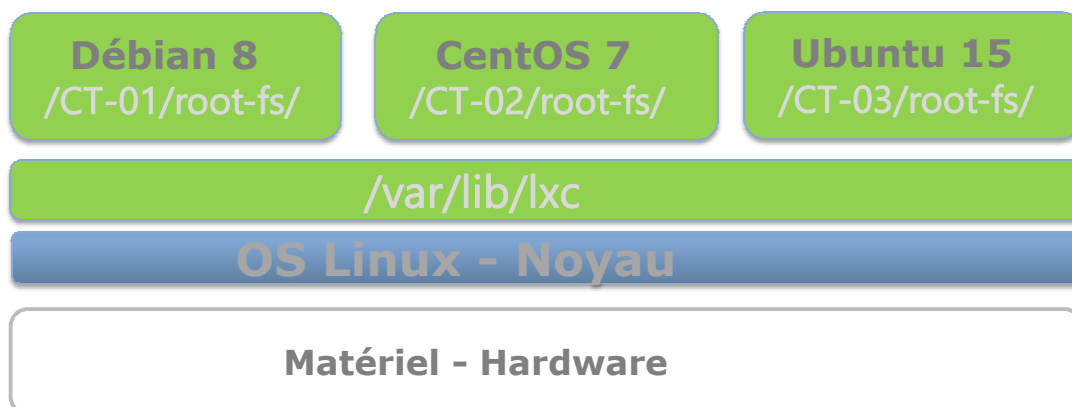
Le modèle : isolation, containers

- L'isolation d'application



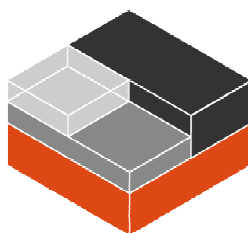
Le modèle : isolation, containers

- Pas de noyau dans les conteneurs, un répertoire égale un OS
- Pas d'émulation de matériel
- Accès direct au matériel



Les produits : "Isolateur", containers

- En isolation :
 - Chroot
 - BSD Jails
 - Les zones de Solaris
- En containers :
 - LXC
 - OpenVZ
 - Linux-Vserver
 - Dockers



Ce qu'on a couvert

- Le modèle : Machines Virtuelles
- Le modèle : isolation, containers
- Les produits : "Isolateur", containers



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Concept

Les Namespaces et les Control groups

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Les espace de noms ou Namespaces
- Création d'un espace de nom réseau
- Introduction aux groupes de contrôle – les cgroups



Les espace de noms ou Namespace

- LXC, OpenVZ, Docker sont des "outils" qui permettent d'isoler très simplement plusieurs petits systèmes d'environnement ou applications sur une machine physique
- Ces environnements exécutent des applications (web, BD, DHCP, ..) qui n'interfèrent ni avec le système installé sur la machine, ni entre eux.
- Ils sont donc **ISOLÉS**
- LXC utilise des fonctions du noyau Linux afin d'isoler ces environnements
- Les Linux namespaces
- Chroot permet une isolation au niveau du système de fichiers
 - Les utilisateurs, les cartes réseaux, les processus. Les noms d'hôtes et noms de domaines ne sont pas isolés du système parent

Les espace de noms ou Namespace

- Les namespaces sont une fonctionnalité du noyau Linux :
 - **Process Namespace**
 - Isole les processus, le conteneur dispose de sa liste de processus
 - **Network Namespace**
 - Isole les interfaces réseau, le conteneur dispose de ses propres interfaces
 - **Mount Namespace**
 - Isole les systèmes de fichiers
 - **UTS Namespace**
 - Permet au conteneur de disposer de ses noms d'hôtes et de domaines
 - **IPC Namespace**
 - Ses propres processus Inter Communications
 - **User Namespace**
 - Permet aux conteneurs de disposer de ses utilisateurs, root dans le conteneur possède l'id et le gid 0

Création d'un espace de nom réseau

- Création du répertoire pour notre espace de nom de type réseau

```
#mkdir /var/run/netns
```

- Création de l'espace de nom de type réseau

```
#ip netns add mon_espace  
#ip netns list  
#ls -l /var/run/netns/
```

- Exécution d'un shell dans l'espace de nom

```
#ip netns exec mon_espace bash  
#ip link list  
1: lo: <LOOPBACK> mtu 65536 qdisc no .....  
#exit  
#ip link list  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default  
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group .
```



Création d'un espace de nom réseau

- Ajout d'une paire d'interface veth00 associé a eth0

```
#ip link add veth00 type veth peer name eth0
#ip link list
#ip address list
```
- Ajout et activation de l'interface eth0

```
#ip address add 10.0.3.10/24 dev eth0
#ip link set eth0 up
```
- Déplace l'interface veth00 dans l'espace de nom de PID 0

```
#ip link set veth00 netns 1
#exit
#ip link set veth00 up
#brctl addif lxcbr0 veth00
#brctl show
#ping 10.0.3.10
```

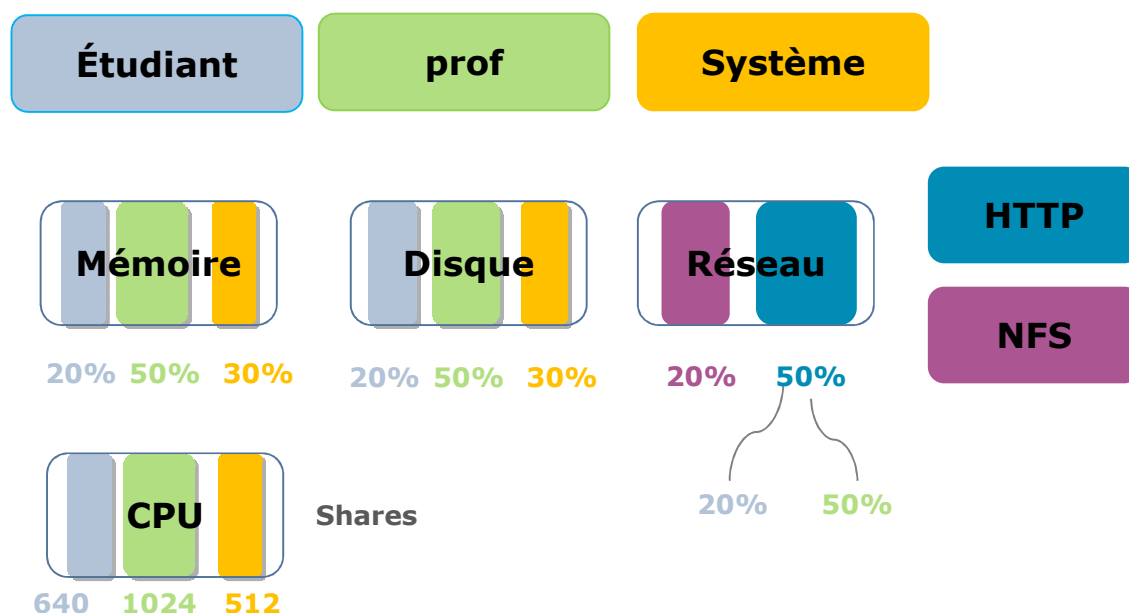


Création d'un espace de nom réseau

Introduction aux groupes de contrôle

- Les **cgroups** ou **Control Groups**- Groupes de contrôle est une fonctionnalité du noyau Linux qui a pour but de :
 - Contrôler les ressources
 - Allocation, l'interdiction, priorisation, figer
 - Surveiller et mesurer les quantités de ressources consommées

Introduction aux groupes de contrôle



Introduction aux groupes de contrôle

- Les cgroups sont organisés en sous-systèmes ou modules
- Un sous-système est un contrôleur de ressources
 - **Blkio** : Surveille et contrôle l'accès des tâches aux entrées/sorties sur des périphériques block
 - **Cpu** : Planifie l'accès de la CPU
 - **Cpuacct** : Rapports sur les CPU utilisées
 - **cpu.shares** : Part relative du temps CPU disponible pour les tâches
 - **Cpuset** : Assigne des CPU à des tâches
 - **Devices** : Autorise ou refuse l'accès aux périphériques
 - **Freezer** : Suspend ou réactive les tâches
 - **memory** : Utilisation mémoire

Ce qu'on a couvert

- Les espace de noms ou Namespace
- Création d'un espace de nom réseau
- Introduction aux groupes de contrôle – les cgroups





Alphorm.com

Concept

LinuxContainers.org
LXC

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Linuxcontainers.org
- Qu'est ce que LXC?
- Architecture des Linux Containers
- LXD – l'hyperviseur pour LXC
- CGManager et LXCFS



Formation Virtualisation légère avec LXC

alphorm.com™©

linuxcontainers.org

- linuxcontainers.org est le projet parent de LXC, LXD, LXCFS et CGManager.
 - L'objectif est d'offrir un environnement neutre, non influencé par des distributions ou fournisseurs, permettant le développement des technologies liées aux conteneurs Linux.
 - Ce projet se concentre sur les conteneurs systèmes. Ceux-ci offrent un environnement aussi proche que possible de celui d'une machine virtuelle, mais sans le surcoût associé à l'utilisation d'un noyau séparé ou à la simulation de matériel.
 - Tout ceci est possible grâce aux fonctionnalités de sécurité du noyau telles que les "**namespaces**", le contrôle d'accès obligatoire (**MAC**) et les groupes de contrôle (**CGroups**).

Qu'est ce que LXC?

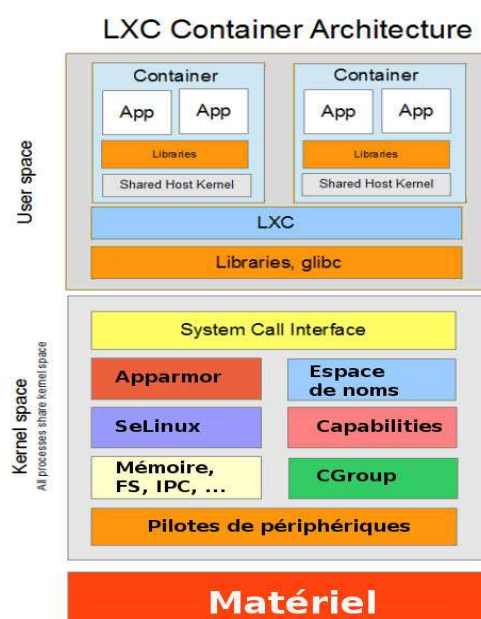
- LXC est une interface en espace utilisateur (**userspace**)
- C'est une API et des outils en licence GNU LGPLv2.1
- Manipuler des conteneurs
- LXC peut être vu comme une solution entre le chroot et une machine virtuel.
- L'idée est de créer et gérer des environnements aussi isolés que possible à partir des noyaux linux standard et ce sans modifications

Architecture des Linux Containers

- LXC pour gérer et créer des conteneurs utilise, des fonctionnalités du noyau Linux :
 - Espace de noms - **namespaces** (*ipc, uts, mount, pid, network and user*)
 - La sécurité MAC Apparmor et SELinux
 - Le Chroot
 - Les capabilities
 - Les groupes de contrôles - CGroups (control groups)

Architecture des Linux Containers

- Outils LXC en User space
- Partage le noyau de l'hôte
- Utilise les fonctionnalités du noyau pour l'isolation et la gestion des processus conteneurs





LXD – l'hyperviseur pour LXC

- LXD est un projet de Canonical Ltd et Ubuntu, c'est une alternative aux outils LXC
- LXD fournit 3 composants :
 - Un daemon LXD (gestion local et distante)
 - Un outil de gestion des containers en CLI
 - Un plugin OpenStack
- LXD n'est pas une nouvelle version de LXC
 - Il est construit au-dessus de LXC et fournit une série d'outils plus simples pour l'utilisateur
 - Offre la sécurité des conteneurs par défaut (non privilégiés, gestion des ressources),
 - Se fier sur un fournisseur d'image (pas de templates)
 - Support de checkpoint et restauration (migration en ligne)



CGManager et LXCFS

- Les groupes de contrôle, les **cgroups** sont traditionnellement manipuler via un système de fichier (/sys/fs/cgroups).
- LXC utilise ces groupes de contrôles pour suivre et limiter les ressources des conteneurs
- Un répertoire propre au conteneur est donc créé et monté pour gérer ces limitations.
- LXC s'appuie sur **Apparmor** pour s'assurer que le conteneur ne monte que son propre cgroups et le suivi des actions dans ses cgroups. Cela rends le code de LXC complexe et volumineux
- Par conséquent, **cgmanager** a été développé pour simplifier la gestion des groupes de contrôle au sein des conteneurs



CGManager et LXCFS

- **Cgmanager** au travers d'une interface **D-Bus** a pour objectif de fournir de manière transparente et sécurisée, l'administration au conteneur ses propres groupes de contrôles
- Les conteneurs peuvent ainsi avec cgmanager monter à l'intérieur de ces derniers, leur cgroups et donc les manipuler
- Cgmanager est particulièrement utile dans le cadre des containers nested et non privilégiés



CGManager et LXCFS

- Le service (daemon) CGManager permet donc de déléguer aux conteneurs et utilisateurs via une interface D-Bus, la gestion simple des cgroups, particulièrement utile pour les "nested containers" et les conteneurs non privilégié.
- Mais quelques programmes continue d'utiliser le fs cgroups, afin de fournir la délégation de ces derniers aux conteneurs, Canonical a développé LXCFS
- Il fournit deux fonctionnalités :
 - Une vue virtualisée de certains fichiers du répertoire `/proc`, `cpu`, `uptime`, `meminfo`. Visible par `/var/lib/lxcfs/proc`
 - Filtre l'accès aux systèmes de fichiers CGroup de l'hôte. `/var/lib/lxcfs/cgroup/`

Ce qu'on a couvert

- linuxcontainers.org
- Qu'est ce que LXC
- Architecture des Linux Containers
- LXD – l'hyperviseur pour LXC
- CGManager et LXCFS



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Mise en œuvre Installation de LXC

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Installation de LXC
- Le package LXC installé
- Vérification de l'installation



Installation de LXC

- Installation du dépôts epel pour Centos 7

```
#wget http://dl.fedoraproject.org/pub/epel/7/x86\_64/e/epel-release-7-5.noarch.rpm
#rpm -ivh epel-release-7-5.noarch.rpm
```
- Recherche des paquets LXC

```
#yum search lxc
```
- Installation des paquets

```
# yum install lxc.x86_64 lxc-templates.x86_64
```
- Installation des paquets

```
#add-apt-repository ppa:ubuntu-lxc/stable
#apt-get install lxc
```



Le package lxc installé

- La configuration :
 - /etc/lxc
 - /etc/default/lxc-net (pont réseau, serveur dhcp)
 - /etc/apparmor.d/lxc
- Les outils :
 - /usr/bin/lxc*
- La documentation :
 - /usr/share/doc/lxc
- Les logs :
 - /var/log/lxc



Le package lxc installé

- Un bridge lxcbr0 est créé
- Un serveur dhcp géré par dnsmasq



Vérification de l'installation

- Vérification de la configuration avec lxc-checkconfig

```
#lxc-checkconfig
```

```
Kernel configuration not found at /proc/config.gz; searching...  
Kernel configuration found at /boot/config-3.10.0-123.el7.x86_64
```

```
--- Namespaces ---
```

```
Namespaces: enabled  
Utsname namespace: enabled  
Ipc namespace: enabled  
Pid namespace: enabled  
User namespace: enabled  
Network namespace: enabled  
Multiple /dev/pts instances: enabled
```

```
--- Control groups ---
```

```
Cgroup: enabled  
Cgroup clone_children flag: enabled  
Cgroup device: enabled  
Cgroup sched: enabled  
Cgroup cpu account: enabled  
Cgroup memory controller: enabled  
Cgroup cpuset: enabled
```

```
--- Misc ---
```

```
Veth pair device: enabled  
Macvlan: enabled  
Vlan: enabled  
File capabilities: enabled
```



Ce qu'on a couvert

- Installation de LXC
- Le package LXC installé
- Vérification de l'installation





Alphorm.com

Mise en œuvre Containers privilégiés

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Un container privilégié ?
- La commande lxc-create
- Où se trouve le conteneur?



Formation Virtualisation légère avec LXC

alphorm.com™©

Un container privilégié ?

- Deux méthodes de création des conteneurs LXC
 - Privilégiés :
 - Le conteneur est créé par l'utilisateur root
 - L'utilisateur root possède l'id 0 dans le conteneur
 - Si via proc, sys ou des appels systèmes, l'utilisateur s'échappe du conteneur, il est alors root sur l'hôte, le conteneur est considéré non sure
 - Non privilégiés :
 - Le conteneur est généralement créé par un utilisateur autre que root, mais peut être créer par le root via les user namespaces

La commande lxc-create

- Création d'un container avec lxc-create

```
#lxc-create -t download --name ubuntu15.10 --dist ubuntu --release wily --arch amd64
#lxc-create -t download --name fedora
#lxc-create -t debian --name debian7 -- -r wheezy --arch amd64
#lxc-create -t centos --name centos7
```

```
-t -template = download : télécharge le modèle du site image.linuxcontainers.org
-t -template = download : interactive si pas de distribution spécifiée
-t -template = debian : par debbootstrap
-t -template = centos : avec yum (doit être installé sur l'hôte : apt-get install yum)
--dist = la distribution utilisée pour le ct
--release = spécifie la version
--arch = l'architecture 32bits/64bits
```

La commande lxc-create

- Création d'un container avec lxc-create

```
#lxc-create -t centos --name centos7
Downloading the image index
Downloading the rootfs
Downloading the metadata
The image cache is now ready
Unpacking the rootfs

---
You just created a CentOS container (release=7, arch=amd64, variant=default)

To enable sshd, run: yum install openssh-server

For security reason, container images ship without user accounts
and without a root password.

Use lxc-attach or chroot directly into the rootfs to set a root password
or create user accounts.
```

Où se trouve le conteneur ?

- Le fichier de configuration
 - /var/lib/lxc/conteneur/config
- Le système de fichiers
 - /var/lib/lxc/conteneur/rootfs/

Ce qu'on a couvert

- Un container privilégié ?
- La commande lxc-create



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Mise en œuvre Containers non-priviliégiés

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Un container non privilégié ?
- Création de l'environnement



Un container non privilégié ?

- Il y a deux méthodes de création des conteneurs LXC :
 - Privilégiés :
 - Le conteneur est créé par l'utilisateur root et il est considéré comme non sécurisé
 - Non privilégiés :
 - Permet à un utilisateur autre que root non privilégié de créer et de gérer ses propres conteneurs lxc
 - Les **namespaces** sont utilisés pour réaliser ses conteneurs "sécurisés"

Un container non privilégié ?

- A partir d'ubuntu 14, chaque utilisateur se voit attribuer, un intervalle d'uid et de gid supplémentaire
- Les fichiers `/etc/subuid` et `/etc/subgid` contiennent ses plages

```
#cat /etc/sub*id
ludo:100000:65536
vicky:200001:65536
ludo:100000:65536
vicky:200001:65536
```

- Ces uid/gid sont utilisés pour créer notre conteneur

```
#grep id_map ~/.config/lxc/default.conf
lxc.id_map = u 0 100000 65536"
lxc.id_map = g 0 100000 65536"
```

Création de l'environnement

- Création de l'environnement pour le conteneur non privilégié

```
#mkdir -p ~/.config/lxc
#vi ~/.config/lxc/default.conf
lxc.id_map = u 0 100000 65536
lxc.id_map = g 0 100000 65536
lxc.network.type = veth
lxc.network.link = lxcbr0

/var/lib/lxc/ = ~/.local/share/lxc
/var/cache/lxc = ~/.cache/lxc

#mkdir -p ~/.local/share/lxc && mkdir ~/.cache/lxc

#echo "$USER veth lxcbr0 2" | sudo tee -a /etc/lxc/lxc-usernet
```

- Création du conteneur

```
#lxc-create -t download -n u1 -- -d ubuntu -r trusty -a amd64
```

Ce qu'on a couvert

- Un container non privilégié ?
- Création de l'environnement



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Exploitation des containers

Commandes
de manipulation
des conteneurs

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Démarrer, arrêter, suspendre, supprimer, lister, ...
- Informations sur les conteneurs
- Commande et console dans les conteneurs



Démarrer, stopper, suspendre, lister les conteneurs

- Démarrer simplement un conteneur
`#lxc-start -name mon_conteneur`
- Démarrer un conteneur en tâche de fond
`#lxc-start -name mon_conteneur --daemon`
- Démarrer un conteneur en tâche de fond et un fichier journal
`#lxc-start -name mon_conteneur --daemon --logfile=/tmp/mon_conteneur.log`
- Démarrer un conteneur en tâche de fond avec un fichier de configuration spécifique
`#lxc-start -name mon_conteneur --daemon -rcfile=/tmp/mon_conteneur.config`
- Démarrage « éphémère » du conteneur `mon_conteneur`, crée un copie de ce dernier
`#lxc-start-ephemeral -o mon_conteneur`
- Démarrage « éphémère » du conteneur `mon_conteneur`, crée un copie de ce dernier, affiche IP et nom de console, "n'attache" pas la console du conteneur
`#lxc-start-ephemeral -o mon_conteneur -n mon_conteneur-ephemeral -d`



Démarrer, stopper, suspendre, lister les conteneurs

- Arrêter proprement un conteneur
`#lxc-stop --name mon_conteneur`
- Redémarrer un conteneur
`#lxc-stop --name mon_conteneur --reboot`
- Tuer les processus a l'intérieur du conteneur
`#lxc-stop --name mon_conteneur --kill`

Valeur de sortie

- 0 Le conteneur s'est arrêté avec succès
- 1 Une erreur lors de l'arrêt du conteneur
- 2 Le conteneur existe mais n'est pas exécuté

- Supprimer un conteneur
`#lxc-destroy --name mon_conteneur -force (attend l'arrêt du conteneur)`



Démarrer, stopper, suspendre, lister les conteneurs

- Suspendre un conteneur
`#lxc-freeze --name mon_conteneur`
- Réactiver le conteneur en pause
`#lxc-unfreeze --name mon_conteneur`
- Monitorer l'état du conteneur (start/stop/thawed/freeze)
`#lxc-monitor -n mon_conteneur`
'mon_conteneur' changed state to [STARTING]
'mon_conteneur' changed state to [RUNNING]
'mon_conteneur' changed state to [FREEZING]
'mon_conteneur' changed state to [FROZEN]
'mon_conteneur' changed state to [THAWED]



Informations sur les conteneurs

- **Information du conteneur**
#lxc-info -n mon_conteneur
Name: mon_conteneur
State: RUNNING
PID: 6989
IP: 10.0.3.39
CPU use: 7.72 seconds
BlkIO use: 664.00 KiB
Memory use: 7.31 MiB
KMem use: 0 bytes
Link: veth1HDL0X
TX bytes: 3.19 KiB
RX bytes: 2.85 KiB
Total bytes: 6.05 KiB



Commande et console dans les conteneurs

- **Exécuter une commande dans le conteneur**
#lxc-attach -n mon_conteneur mkdir /root/test.attach
- **Exécuter le shell dans le conteneur (permet de « rentrer » dans ce dernier)**
#lxc-attach -n mon_conteneur
root@mon_conteneur:~# ls -l /root/
drwxr-xr-x 2 root root 4096 Aug 3 12:52 test.attach/
- **Entrer en mode console dans le conteneur**
#lxc-console -n mon_conteneur
Connected to tty 1
Type <Ctrl+a q> to exit the console, <Ctrl+a Ctrl+a> to enter Ctrl+a itself

Ubuntu 14.04.2 LTS mon_conteneur tty1
mon_conteneur login:

Ce qu'on a couvert

- Démarrer, arrêter, suspendre, supprimer, lister, ...
- Informations sur les conteneurs
- Commande et console dans les conteneurs



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Exploitation des containers

Le clonage des conteneurs

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Introduction au clonage
- Les clones "complets"
- Les clones avec snapshot



Introduction au clonage

- Copier la totalité d'un conteneur
 - Fichier de configuration
 - Le repertoire rootfs
- Pas seulement !
 - Modification du fichier de configuration :
 - namespaces, hostname, configuration réseau, adresse mac
- La taille est sensiblement identique
- Clone lié ou clone avec snapshot
 - Très rapide
 - Très léger



Les clones "complets"

- **Simple clone d'un conteneur**
`#lxc-clone --orig mon_conteneur --new mon_conteneur_cloné`

`lxc-clone [-s] [-B backingstore] [-L size[unit]] [-K] [-M] [-H]
[-p lxcpath] [-P newlxcpath] orig new`

-B: Spécifie un Backends (systemen de fichiers), ZFS, LVM, BTRFS, fs local

-L : La taille en Mo
-K : Conserve le nom du conteneur
-M : Conserve l'adresse MAC original
-p : lxcpath, le chemin du conteneur a cloner
-P : lxcpath du nouveau conteneur, par default /var/lib/lxc.



Les clones avec snapshots

- **Simple clone d'un conteneur**
`#lxc-clone --orig mon_conteneur --new mon_conteneur_cloné -s`

`lxc-clone [-s] [-B backingstore] [-L size[unit]] [-K] [-M] [-H]
[-p lxcpath] [-P newlxcpath] orig new`

-B: Spécifie un Backends (systemen de fichiers), ZFS, LVM, BTRFS, fs local

-L : La taille en Mo
-K : Conserve le nom du conteneur
-M : Conserve l'adresse MAC original
-p : lxcpath, le chemin du conteneur a cloner
-P : lxcpath du nouveau conteneur, par default /var/lib/lxc.

Ce qu'on a couvert

- Introduction au clonage
- Les clones "complets"
- Les clones avec snapshot



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Exploitation des containers

Snapshot

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Introduction au snapshot
- Snapshot avec LXC



Introduction au snapshot

- Un snapshot est une image du système de fichiers ou d'un de ses sous-volumes à un instant T
- Une fois créé, un snapshot se comporte comme un dossier ordinaire de votre système de fichiers
- Vous pouvez le copier, le déplacer vers un support externe, le renommer...

Snapshot avec LXC

- **Snapshot simple d'un conteneur**
`#lxc-snapshot -n mon_conteneur`

`lxc-snapshot snapshots un conteneur`

Options :
 - n, --name=NAME NAME nom du conteneur
 - L, --list La liste des snapshots
 - C, --showcomments Affiche les commentaires du snapshot
 - c, --comment=file Ajout de commentaire via fichier
 - r, --restore=name restaure un snapshot de nom 'snap0'
 - d, --destroy=name supprime le snapshot 'snap0'
- **Restauration d'un snapshot**
`#lxc-snapshot -n samba -r snap0`

Ce qu'on a couvert

- Introduction au snapshot
- Snapshot avec LXC





Alphorm.com

Exploitation des containers avec LXC

Gestion des ressources avec les cgroup

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- CGManager et LXCFS
- Limiter la mémoire
- Limiter le nombre de processeur
- Gerer les partage CPU



Formation Virtualisation légère avec LXC

alphorm.com™©



CGManager et LXCFS

- Les groupes de contrôle, les **cgroups** sont traditionnellement manipuler via un système de fichier (`/sys/fs/cgroups`).
- LXC utilise ces groupes de contrôles pour suivre et limiter les ressources des conteneurs
- Un répertoire propre au conteneur est donc créé et monté pour gérer ces limitations.
- Puisque LXC s'appuie sur **Apparmor** pour s'assurer que le conteneur ne monte que son propre cgroups et s'assure du suivi des actions dans ses cgroups. Cela rends le code de LXC complexe et volumineux
- Par conséquent, **cgmanager** a été développé pour simplifier la gestion des groupes de contrôle au sein des conteneurs



CGManager et LXCFS

- Au travers d'une interface **D-Bus**, **Cgmanager** a pour objectif de fournir de manière transparente et sécurisée, l'administration au conteneur ses propres groupes de contrôles
- Les conteneurs peuvent ainsi avec cgmanager monter à l'intérieur de ces derniers, leur cgroups et donc les manipuler
- Cgmanager est particulièrement utile dans le cadre des containers nested et non privilégiés



CGManager et LXCFS

- Quelques programmes continuent d'utiliser le fs cgroups, afin de fournir la délégation de ces derniers aux conteneurs, Canonical a développé LXCFS
- Il fournit deux fonctionnalités :
 - Une vue virtualisée de certains fichiers du répertoire `/proc`, `cpu`, `uptime`, `meminfo`. Visible via `/var/lib/lxcfs/proc`
 - Filtre l'accès aux systèmes de fichiers CGroup de l'hôte. `/var/lib/lxcfs/cgroup/`



Cgroups mémoire

- Les groupes de contrôles cpu sur le conteneur netservice

```
root@net service:~# ls -l /sys/fs/cgroup/memory/lxc/net service/  
memory.limit_in_bytes  
memory.soft_limit_in_bytes  
memory.max_usage_in_bytes
```

```
root@node1# lxc-cgroup -n net service memory.limit_in_bytes 511888000  
root@net service:~# echo 511888 > /sys/fs/cgroup/memory/lxc/net service/  
memory.limit_in_bytes  
root@net service:~# cat /sys/fs/cgroup/memory/lxc/net service/memory.stat  
root@node1# vi /var/lib/lxc/net service/config  
lxc.cgroup.memory.limit_in_bytes = 512M
```



Cgroups sur les processeur

- Les groupes de controles mémoire sur le conteneur netservice

```
root@netservice:~# ls -l /sys/fs/cgroup/cpuset/lxc/netservice/  
cpuset.cpus  
root@netservice:~# cat /sys/fs/cgroup/cpu/lxc/netservice/cpu.shares  
256  
  
root@node1# lxc-cgroup -n netservice cpuset.cpus 0  
root@node1# lxc-cgroup -n netservice cpu.shares 512  
  
root@netservice:~# echo 0 > /sys/fs/cgroup/cpuset/lxc/netservice/  
cpuset.cpus  
root@netservice:~# cat /sys/fs/cgroup/cpuacct/lxc/netservice/cpuacct.stat  
root@node1# vi /var/lib/lxc/netservice/config  
lxc.cgroup.cpuset.cpus = 0  
lxc.cgroup.cpu.shares = 512
```



Ce qu'on a couvert

- CGManager et LXCFS
- Cgroups mémoire
- Cgroups sur les processeur





Alphorm.com

Exploitation
des containers avec LXD

LXD, présentation et installation

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Introduction à LXD
- Installation de LXD

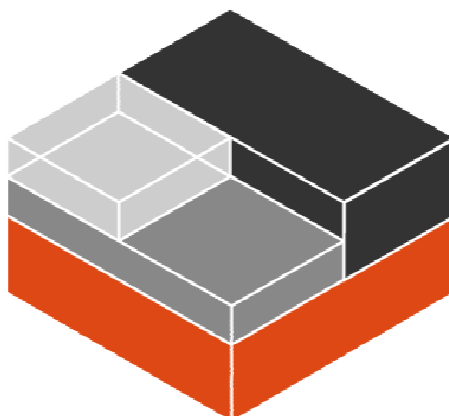


Formation Virtualisation légère avec LXC

alphorm.com™©

Introduction a LXD

- LXD est un projet fondé et maintenu par Canonical Ltd, LXD n'est pas une nouvelle version de LXC



Introduction a LXD

- LXD fournit :
 - La gestion des images
 - La gestion des conteneurs
 - La gestion de multiples serveurs distant
 - La migration de conteneur en ligne

Introduction a LXD

- LXD comprends trois composants :
 - Un démon (service) qui fournit une API REST (local, distant)
 - Un outil en ligne de commande LXC
 - Un plug-in pour Openstack

Installation de LXD

- Installation

```
#add-apt-repository ppa:ubuntu-lxc/lxd-stable
#apt-get update
#apt-get install lxd-client lxd
```
- Installation de criu (checkpoint pour les migrations live)

```
#apt-get install criu
```

Ce qu'on a couvert

- Introduction à LXD
- Installation de LXD



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Exploitation des containers avec LXD

Gestion des images LXD

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©



Plan

- Introduction à images.linuxcontainers.org
- Manipuler des images
- Publier des images



Introduction à images.linuxcontainers.org

- Dépôts d'images pour LXD :

← → ↺ <https://images.linuxcontainers.org/images/>

Name	Last modified	Size	Description
Parent Directory		-	
centos/	2014-07-09 03:27	-	
debian/	2015-05-12 00:20	-	
fedora/	2015-07-19 06:43	-	
gentoo/	2014-01-28 11:15	-	
opensuse/	2014-06-25 22:40	-	
oracle/	2014-01-15 18:12	-	
plamo/	2014-01-14 20:28	-	
ubuntu/	2015-05-12 04:50	-	

Apache/2.4.7 (Ubuntu) Server at images.linuxcontainers.org Port 80

 Manipuler des images

- Récupérer des images

```
#lxd-images import ubuntu trusty amd64 --alias ubuntu/trusty
images de type Ubuntu

#lxd-images import lxc debian wheezy amd64 --alias debian/wheezy
images de type LXC
```
- Lister les images

```
#lxc image list
```

ALIAS	FINGERPRINT	PUBLIC	DESCRIPTION	ARCH	UPLOAD DATE
debian/wheezy	0fce9c160501	no	debian wheezy x86_64 (default) (20150816_22:42)	x86_64	Aug 11, 2015 at 11:50am
ubuntu/trusty	66cbdd1ba9af	no	Ubuntu 14.04 LTS server (20150817)	x86_64	Aug 11, 2015 at 11:36am

 Manipuler des images

- Visualiser des images

```
# lxc image show debian/wheezy
architecture: x86_64
description: debian wheezy x86_64 (default) (20150816_22:42)
name: debian-wheezy-x86_64-default-20150816_22:42
os: debian
release: wheezy
variant: default
```



Manipuler des images

- **Information d'une image**

```
#lxc image info debian/wheezy
```

```
Fingerprint: 0fce9c160501cd71d8600de4e3053a091c876ff3be0aef71bb4babbb35ccb9e9
```

```
Size: 93MB
```

```
Architecture: x86_64
```

```
Public: no
```

```
Timestamps:
```

```
Created: 2015/08/16 22:42 UTC
```

```
Uploaded: 2015/08/11 09:50 UTC
```

```
Expires: never
```

```
Properties:
```

```
os: debian
```

```
release: wheezy
```

```
variant: default
```

```
architecture: x86_64
```

```
description: debian wheezy x86_64 (default) (20150816_22:42)
```

```
name: debian-wheezy-x86_64-default-20150816_22:42
```

```
Aliases:
```

```
- debian/wheezy
```

- **Éditer les information d'une image**

```
#lxc image edit debian/wheezy
```



Manipuler des images

- **Ajout d'un serveur d'image distant**

```
#lxc remote add srv-images images.linuxcontainers.org
```

```
#lxc remote list
```

```
srv-images https://images.linuxcontainers.org:8443
```

- **Lister les images du serveur distant**

```
#lxc image list srv-images:
```

ALIAS	FINGERPRINT	PUBLIC	DESCRIPTION	ARCH	UPLOAD DATE
	87a2276c8479	yes	Centos 6 (amd64)	x86_64	Aug 15, 2015 at 5:17am (CEST)
	a8c12e49dc05	yes	Centos 6 (amd64)	x86_64	Aug 16, 2015 at 8:17pm (CEST)
centos/6/amd64 (1 more)	dba7b2e673da	yes	Centos 6 (amd64)	x86_64	Aug 17, 2015 at 8:17pm (CEST)
	876d009e3ab2	yes	Centos 6 (i386)	i686	Aug 15, 2015 at 5:19am (CEST)
	f0572140edde	yes	Centos 6 (i386)	i686	Aug 16, 2015 at 8:19pm (CEST)
centos/6/i386 (1 more)	68ac8b441c1c	yes	Centos 6 (i386)	i686	Aug 17, 2015 at 8:19pm (CEST)
	027cb2473faf	yes	Centos 7 (amd64)	x86_64	Aug 15, 2015 at 5:22am (CEST)
	36dce44b9918	yes	Centos 7 (amd64)	x86_64	Aug 16, 2015 at 8:22pm (CEST)
centos/7/amd64 (1 more)	02b7ebe8a583	yes	Centos 7 (amd64)	x86_64	Aug 17, 2015 at 8:22pm (CEST)



Manipuler des images

- Ajout d'un serveur d'image distant

```
#lxc remote add node2 node2.alphorm.local:8443
```

```
#lxc image list node2:
```

ALIAS	FINGERPRINT	PUBLIC	DESCRIPTION	ARCH	UPLOAD DATE
debian/wheezy	0fce9c160501	no	debian wheezy x86_64 (default)	x86_64	Aug 11, 2015 at 11:50am
ubuntu/trusty	66cbdd1ba9af	no	Ubuntu 14.04 LTS server	x86_64	Aug 11, 2015 at 11:36am

- Publier une image

```
#lxc launch node2:ubuntu/trusty/amd64 mon_ct
```

```
#lxc publish --public mon_ct --alias=trusty_modif_by_ludo
```

```
Container published with fingerprint
```

```
03e61b2a7b9c895c6a322bcc29ac334a60318d33f8a7d58cd70768f179bde8d4
```

```
#lxc image list
```

ALIAS	FINGERPRINT
trusty_modif_by_ludo	03e61b2a7b9c



Ce qu'on a couvert

- Introduction à images.linuxcontainers.org
- Manipuler des images
- Publier des images





Alphorm.com

Exploitation
des containers avec LXD

Manipulation des conteneurs

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Créer ses conteneurs avec lxc launch et lxc init
- Démarrer, arrêter, redémarrer des conteneurs
- Ouvrir une console
- Obtenir des informations sur les conteneurs
- Destruction des conteneurs



Formation Virtualisation légère avec LXC

alphorm.com™©

Créer ses conteneurs

- Création d'un conteneur à partir d'une image

```
#lxc launch srv-images:ubuntu/wily/amd64 mon_ct
Creating mon_ct done.
Starting mon_ct done.
```

```
#lxc init images-lxc-org:ubuntu/wily/amd64 mon_ct2
Creating mon_ct2 done.
```

- Lister les conteneurs

```
#lxc list
+-----+-----+-----+-----+-----+-----+
| NAME   | STATE | IPV4   | IPV6   | EPHEMERAL | SNAPSHOTS |
+-----+-----+-----+-----+-----+-----+
| mon_ct | RUNNING | 10.0.3.3 |      | NO         | 1          |
| mon_ct2 | STOPPED |      |      | NO         | 0          |
```

Ce qu'on a couvert

- Créer ses conteneurs avec lxc launch et lxc init
- Démarrer, arrêter, redémarrer des conteneurs
- Ouvrir une console
- Obtenir des informations sur les conteneurs
- Destruction des conteneurs





Alphorm.com

Exploitation
des containers avec LXD

Les snapshots

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Introduction aux snapshots avec LXD
- Prendre des snapshots
- Restaurer et supprimer des snapshots



Formation Virtualisation légère avec LXC

alphorm.com™©

Introduction aux snapshots avec LXD

- Un Snapshot ou instantané est une capture de l'état du conteneur à un instant T (Mémoire et disque).
- Un instantané ou snapshot permet de créer des points de restauration
- LXD permet des snapshots et restauration en ligne

Prendre des snapshots

- Création d'un instantané du conteneur mon_ct

```
#lxc snapshot mon_ct snap_freshinstall
#lxc info mon_ct
Name: mon_ct
Status: RUNNING
Init: 9311
Ips:
Snapshots:
  mon_ct/snap_freshinstall

#lxc snapshot mon_ct snap_freshinstall
....
Snapshots:
  mon_ct/snap_freshinstall
  mon_ct/snap1
```



Restaurer et supprimer des snapshots

- Restauration du snapshot snap_freshinstall

```
#lxc restore mon_ct snap_freshinstall
```

- Suppression du snapshot snap_freshinstall

```
#lxc delete mon_ct/snap_freshinstall
```



Ce qu'on a couvert

- Introduction aux snapshots avec LXD
- Prendre des snapshots
- Restaurer et supprimer des snapshots





Alphorm.com

Exploitation
des containers avec LXD

Administration
des serveurs distants

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- LXD hyperviseur ou "orchestrateur"
- Configuration des serveurs
- Ajout d'un serveur en local
- Ajout des serveurs distants
- Administrer les serveurs



Formation Virtualisation légère avec LXC

alphorm.com™©

LXD hyperviseur ou "orchestrateur"

- Canonical développe LXD
- Canonical a nommé LXD : l'hyperviseur
- LXD peut être vu, plus comme un Orchestrator, un système d'administration que comme un hyperviseur.
- LXD permet d'administrer des conteneurs
- LXD permet d'administrer des serveurs d'images locaux et distants
- LXD permet d'administrer des serveurs distants

Configuration des serveurs

- Configuration d'un mot de passe
`#lxc config set core.trust_password MON_PASSWORD`
- Configuration de l'api distant
`#lxc config set core.https_address @IP`



Ajout d'un serveur en local

- Ajout de l'api

```
#lxc remote add node1 node1:8443
```

```
Certificate fingerprint: 4e 6a ae 2e e9 f4 e5 7b ba 4a a9 cf 82 32 ac 29 03 b3 39 9e 23  
6e cf 90 fb 77 b7 b2 a8 63 49 65
```

```
ok (y/n)? y
```

```
Admin password for node1:
```

```
Client certificate stored at server: node1
```

```
#lxc remote list
```

```
node1 <https://node1:8443>
```

```
local <unix:///var/lib/lxd/unix.socket>
```



Ajout des serveurs distants

- Ajout de l'api

```
root@node1#lxc remote add node2 node2:8443
```

```
Certificate fingerprint: 4e 6a ae 2e e9 f4 e5 7b ba 4a a9 cf 82 32 ac 29 03 b3 39 9e 23  
6e cf 90 fb 77 b7 b2 a8 63 49 65
```

```
ok (y/n)? y
```

```
Admin password for node1:
```

```
Client certificate stored at server: node1
```

```
root@node1#lxc remote list
```

```
node1 <https://node1:8443>
```

```
node2 <https://node2:8443>
```

```
local <unix:///var/lib/lxd/unix.socket>
```

- Ajout d'un serveur d'images

```
root@node1#lxc remote add srv_images images.linuxcontainers.org
```

LXC remote

- Lister les serveurs

```
root@node1#lxc config trust list
```

```
+-----+-----+-----+-----+
| FINGERPRINT | COMMON NAME |   ISSUE DATE   |   EXPIRY DATE   |
+-----+-----+-----+-----+
| 656216f60149 | root@node1 | Aug 11, 2015 at 9:30am (UTC) | Aug 8, 2025 at 9:30am (UTC) |
| ec75b2854712 | root@node2 | Aug 11, 2015 at 9:55am (UTC) | Aug 8, 2025 at 9:55am (UTC) |
+-----+-----+-----+-----+
```

- Sélection du serveur par défaut

```
root@node1#lxc remote set-default node1
root@node1#lxc remote get-default
node1
```

- Supprimer un serveur distant

```
root@node1#lxc remote remove node3
```

Administrer les serveurs

- Créer un conteneur sur le serveur node2

```
root@node1#lxc launch images:ubuntu/trusty/amd64 node2:dc-samba
```

- Copie d'un conteneur entre serveurs

```
root@node1#lxc copy node2:dc-samba node1:bdc-samba
```

- Éditions du fichier de configuration du CT dc-samba sur le serveur node2

```
root@node1#lxc config edit node2:dc-samba
```

- Supprimer un snapshot snap0 du conteneur web-srv sur le serveur distant node3

```
root@node1#lxc delete node3:web-srv/snap0
```

Ce qu'on a couvert

- LXD hyperviseur ou "orchestrateur"
- Configuration des serveurs
- Ajout d'un serveur en local
- Ajout des serveurs distants
- Administrer les serveurs



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Exploitation des containers avec LXD

Migration des conteneurs

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Principe et pré-requis
- Configuration des conteneurs
- Migration des conteneurs



Principe et pré-requis

- La migration consiste à déplacer des conteneurs entre différents hôtes
- LXD propose la migration en ligne, LXD déplace les conteneurs entre différents hôtes pendant leur fonctionnement.
- LXD utilise CRIU pour gérer l'état des conteneurs pendant la migration
http://criu.org/Main_Page
- Criu doit donc être installé sur les hôtes
- Un profil spécifique doit être appliqué au conteneur



Configuration des conteneurs

- Installation de CRIU

```
#apt-get install criu
```

- Création d'un profile dans la BD LXD

```
root@node1#lxc profile create migrate_ct
```

- Éditions du fichier de configuration du profile

```
root@node1#lxc profile edit create migrate_ct
name: migrate_ct
config: raw.lxc: |
    lxc.console = none
    lxc.cgroup.devices.deny = c 5:1 rwm
    lxc.start.auto = lxc.start.auto = proc:mixed sys:mixed
security.privileged: "true"
devices: {}
```



Configuration des conteneurs

- On applique le profile au conteneur

```
root@node1#lxc profile apply mon_ct migrate_ct
```

- On copie le profile sur les différents hôtes

```
root@node1#lxc profile copy mon_ct migrate_ct node2:
```

- Au besoin lxc remote add

```
root@node1#lxc remote add node2 node2:8443
root@node2#lxc remote add node1 node1:8443
```

Migration des conteneurs

- Lister les conteneurs

```
root@node1#lxc list
+-----+-----+-----+
| NAME | STATE | IPV4 |
+-----+-----+-----+
| mon_ct | RUNNING | 10.0.3.73 |
```

- Migration du conteneur mon_ct

```
root@node1#lxc move node1: mon_ct node2: mon_ct
```

- Lister les conteneurs

```
root@node2#lxc list
+-----+-----+-----+
| NAME | STATE | IPV4 |
+-----+-----+-----+
| mon_ct | STOPPED |

```

Ce qu'on a couvert

- Principe et pré-requis
- Configuration des conteneurs
- Migration des conteneurs





Alphorm.com

Configuration avancée
des containers

Le modèle réseau

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Le modèle réseau par défaut : le mode NAT
- Les différents modes de connexion : Le mode empty
- Les différents modes de connexion : Le mode bridge veth
- Les différents modes de connexion : Le mode Macvlan
- Les différents modes de connexion : Le mode phys



Formation Virtualisation légère avec LXC

alphorm.com™©

Le modèle réseau par défaut

- Lors de la création d'un conteneur avec la `lxc-create` ou `lxc launch` (lxd)

- Création d'un conteneur de type Debian

```
#lxc-create -t debian -n mon_conteneur -r wheezy -a amd64
#lxc launch serveur_images:debian/wheezy/amd64 mon_conteneur
```

- Le mode de connexion par défaut est defini par le fichier :

- Un bridge et un veth

```
#cat /etc/lxc/default.conf
lxc.network.type = veth
lxc.network.link = lxcbr0
lxc.network.flags = up
lxc.network.hwaddr = 00:16:3e:xx:xx:xx
```

Le modèle réseau par défaut

- Lors de la création d'un conteneur avec la `lxc-create` ou `lxc launch` (lxd)

- Création d'un conteneur de type Debian

```
#lxc-create -t debian -n mon_conteneur -r wheezy -a amd64
#lxc launch serveur_images:debian/wheezy/amd64 mon_conteneur
```

- Chaque conteneur possède son fichier de configuration

- Un bridge et un veth

```
#cat /var/lib/lxc/mon_conteneur/config
lxc.network.type = veth
lxc.network.link = lxcbr0
lxc.network.flags = up
lxc.network.hwaddr = 00:16:3e:b3:3a:71
```

Le modèle réseau par défaut

- Le conteneur est ponté sur `lxcbr0` et une carte `vethXXXX`

```

bridge name   bridge id           STP enabled  interfaces
lxcbr0        8000.fe5729fd41a7    no           vethV7JMVl

#ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default
   inet 127.0.0.1/8 scope host lo
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
   inet 192.168.1.100/24 brd 192.168.1.255 scope global eth0
3: lxcbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
link/ether 3e:37:33:cc:9a:c3 brd ff:ff:ff:ff:ff:ff
   inet 10.0.3.1/24 scope global lxcbr0
vethV7JMVl: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master lxcbr0 state UP group default qlen 1000
   link/ether fe:57:29:fd:41:a7 brd ff:ff:ff:ff:ff:ff

```

Le modèle réseau par défaut

- Un bridge `lxcbr0` a donc été créé à l'installation de LXC avec pour adresse IP `10.0.3.1/24`, le conteneur possède un adresse en `10.0.3.x/24`

```

#ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default
   inet 127.0.0.1/8 scope host lo
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
   inet 192.168.1.100/24 brd 192.168.1.255 scope global eth0
3: lxcbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
link/ether 3e:37:33:cc:9a:c3 brd ff:ff:ff:ff:ff:ff
   inet 10.0.3.1/24 scope global lxcbr0

#lxc-ls --fancy
NAME      STATE      IPV4
-----
Mon_ct    RUNNING    10.0.3.32

```

Le modèle réseau par défaut

- Le conteneur possède une carte réseau eth0 et accède à l'extérieur du réseau avec **dnsmasq** (petit serveur DNS, DHCP) et **iptables**

```
root@mon_ct:~# ifconfig
eth0  Link encap:Ethernet  HWaddr 00:16:3e:b3:3a:71
      inet addr:10.0.3.32  Bcast:10.0.3.255  Mask:255.255.255.0

#cat /etc/default/lxc-net
USE_LXC_BRIDGE="true"
LXC_BRIDGE="lxcbr0"
LXC_ADDR="10.0.3.1"
LXC_NETMASK="255.255.255.0"
LXC_NETWORK="10.0.3.0/24"
LXC_DHCP_RANGE="10.0.3.2,10.0.3.254"
LXC_DHCP_MAX="253"

#iptables -t nat -L
Chain POSTROUTING (policy ACCEPT)
target  prot opt source      destination
MASQUERADE  all  --  10.0.3.0/24  !10.0.3.0/24
```

Le mode empty

- Le container ne possède pas de carte réseau, pas d'adresse ip !

```
root@hote#vi /var/lib/lxc/mon_ct/config
#lxc.network.type = veth
lxc.network.type = empty
#lxc.network.link = lxcbr0
lxc.network.flags = up
lxc.network.hwaddr = 00:16:3e:b3:3a:71

root@mon_ct#ifconfig
lo  Link encap:Local Loopback
    inet addr:127.0.0.1  Mask:255.0.0.0
    inet6 addr: ::1/128 Scope:Host

root@hote#ifconfig
eth0  Link encap:Ethernet  HWaddr 08:00:27:36:5a:0c
      inet addr:192.168.1.100  Bcast:192.168.1.255  Mask:255.255.255.0
lo  Link encap:Local Loopback
    inet addr:127.0.0.1  Mask:255.0.0.0
lxcbr0  Link encap:Ethernet  HWaddr 00:00:00:00:00:00
      inet addr:10.0.3.1
```

Le mode bridge, veth

- Ce type de connexion réseau (mode par défaut) fournit un périphérique réseau de type "peer"
- Le périphérique "peer" peut être vu comme une paire de périphériques qui agit comme un "tube"
- Ces périphériques ne sont pas des périphériques point à point, ce sont des périphériques de types ethernet qui permettent le broadcast (diffusion)
- Lors de la création du conteneur un périphérique veth est créé sur l'hôte et un périphérique eth0 dans le conteneur
- La partie hôte du périphérique est ponté sur un bridge br0 par exemple
- Le conteneur accède à internet et aux autres machines du réseau, il peut se trouver sur le même segment réseau

Le mode bridge, veth : Configuration

- Désactivation du mode bridge lxcbr0

```
# cat /etc/default/lxc-net  
USE_LXC_BRIDGE="false"
```

- Création d'un bridge br0

```
root@hote# cat /etc/network/interfaces  
...  
auto br0  
iface br0 inet static  
bridge_ports eth0  
bridge_fd 0  
address 192.168.1.11  
netmask 255.255.255.0  
network 192.168.1.0  
broadcast 192.168.1.255  
gateway 192.168.1.1  
dns-nameservers 8.8.8.8
```



Le mode bridge, veth : Configuration

- Désactivation du mode bridge lxcbr0

```
#cat /var/lib/lxc/mon_ct_bridge/config
# Common configuration
lxc.include = /usr/share/lxc/config/ubuntu.common.conf

# Container specific configuration
lxc.rootfs = /var/lib/lxc/mon_ct_bridge/rootfs
lxc.utsname = mon_ct_bridge
lxc.arch = i686

# Network configuration
lxc.network.type = veth
lxc.network.link = br0
lxc.network.flags = up
lxc.network.hwaddr = 00:16:3e:bf:11:52
lxc.network.ipv4 = 192.168.1.110/24
lxc.network.ipv4.gateway = 192.168.1.1
```



Le mode Macvlan

- Le périphérique de type Macvlan est un module Linux
- A partir d'une seule interface réseau (one to many)
- Il fournit plusieurs interfaces virtuelles avec leur propre adresse MAC (many to one)
- Macvlan fournit l'isolation des interfaces virtuelles
- Beaucoup plus léger que le mode Bridge
- Pas de mode promiscus
- Pas de carte Veth
- Plusieurs modes disponibles :
 - Private, VEPA , Bridge

Le mode Macvlan : Configuration

- Désactivation du mode bridge lxcbr0

```
#cat /var/lib/lxc/mon_ct_macvlan/config
# Common configuration
lxc.include = /usr/share/lxc/config/ubuntu.common.conf

# Container specific configuration
lxc.rootfs = /var/lib/lxc/mon_ct_macvlan/rootfs
lxc.utsname = mon_ct_macvlan
lxc.arch = i686

# Network configuration
lxc.network.type = macvlan
lxc.network.macvlan.mode = bridge
lxc.network.link = br0
lxc.network.flags = up
lxc.network.hwaddr = 00:16:3e:bf:11:52
lxc.network.ipv4 = 192.168.1.110/24
lxc.network.ipv4.gateway = 192.168.1.1
```

Le mode phys

- Le périphérique de type phys
- Déplace la carte physique de l'hôte (eth1) vers le conteneur
- L'interface physique est déplacée d'un espace de nom vers un autre espace de nom (le conteneur)
- L'interface n'est plus disponible pour l'hôte, se trouve dans le conteneur

- Configuration du périphériques PHYS

```
#cat /var/lib/lxc/mon_ct_phys/config
lxc.network.type = phys
lxc.network.link = eth1
lxc.network.flags = up
lxc.network.hwaddr = 00:16:3e:bf:11:52
lxc.network.ipv4 = 192.168.1.110/24
lxc.network.ipv4.gateway = 192.168.1.1
```

Ce qu'on a couvert

- Le modèle réseau par défaut
- Les différents modes de connexion : Le mode empty
- Les différents modes de connexion : Le mode bridge veth
- Les différents modes de connexion : Le mode Macvlan
- Les différents modes de connexion : Le mode phys



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Configuration avancée des containers

OpenvSwitch et LXC

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

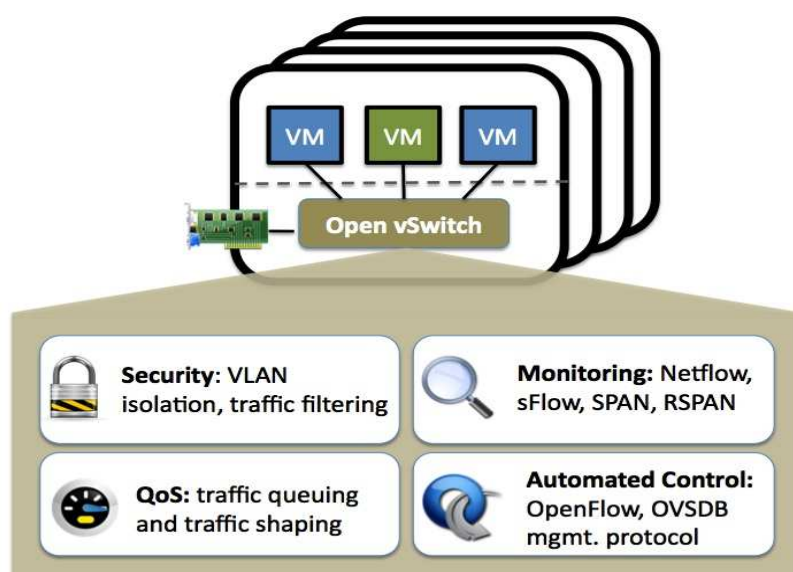
alphorm.com™©

Plan

- Présentation d'OpenvSwitch
- Architecture d'OpenvSwitch
- Configuration basique
- Configuration de LXC et des conteneurs



Présentation OpenvSwitch





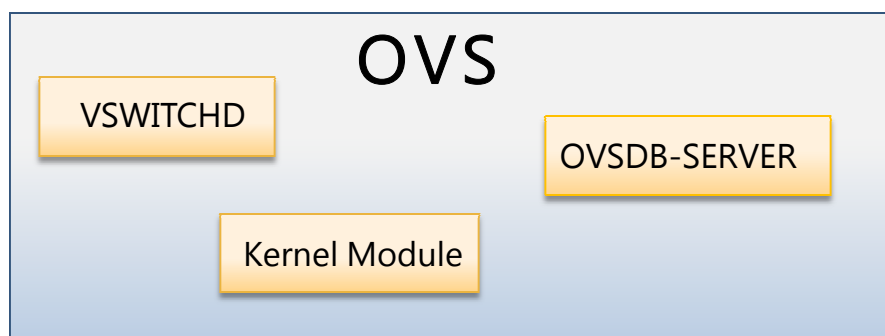
Présentation d'OpenvSwitch

- OpenvSwitch est un commutateur virtuel compatible avec les chipsets des switchs modernes
- Commutateur administrable avec le protocole OpenFlow
- Supporte le VLAN 802.1Q, isolation et filtre de traffics
- Agrégation de lien, lacp, channel bonding
- Gestion des flux, QoS Bande passante
- Collecte de données avec NetFlow, sFlow, RSPAN, SPAN (mirror)

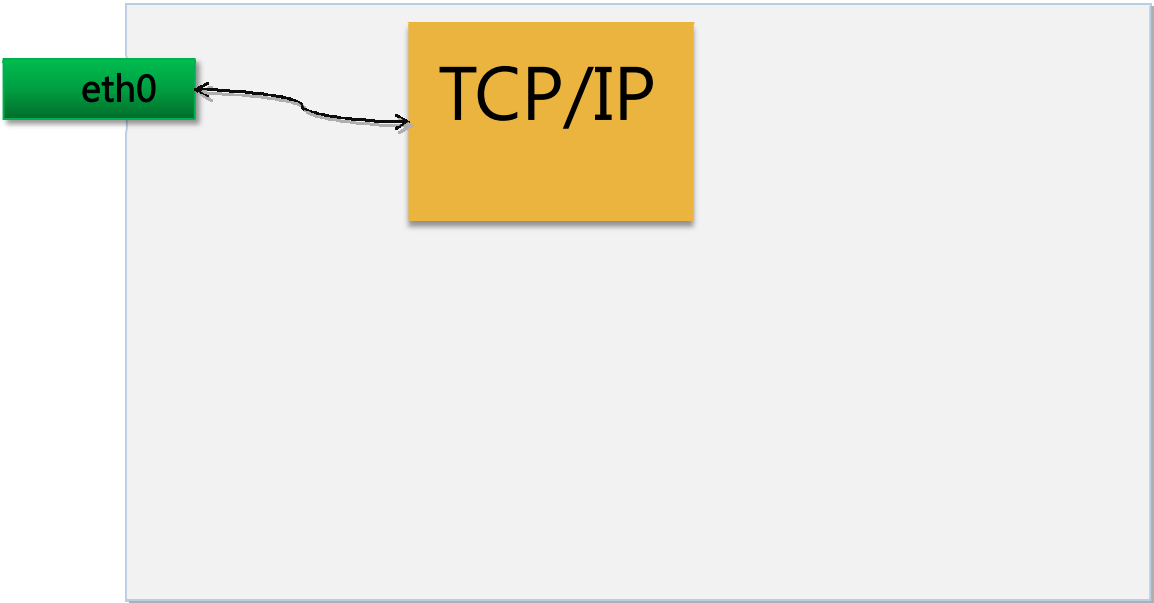


Architecture d'OpenvSwitch

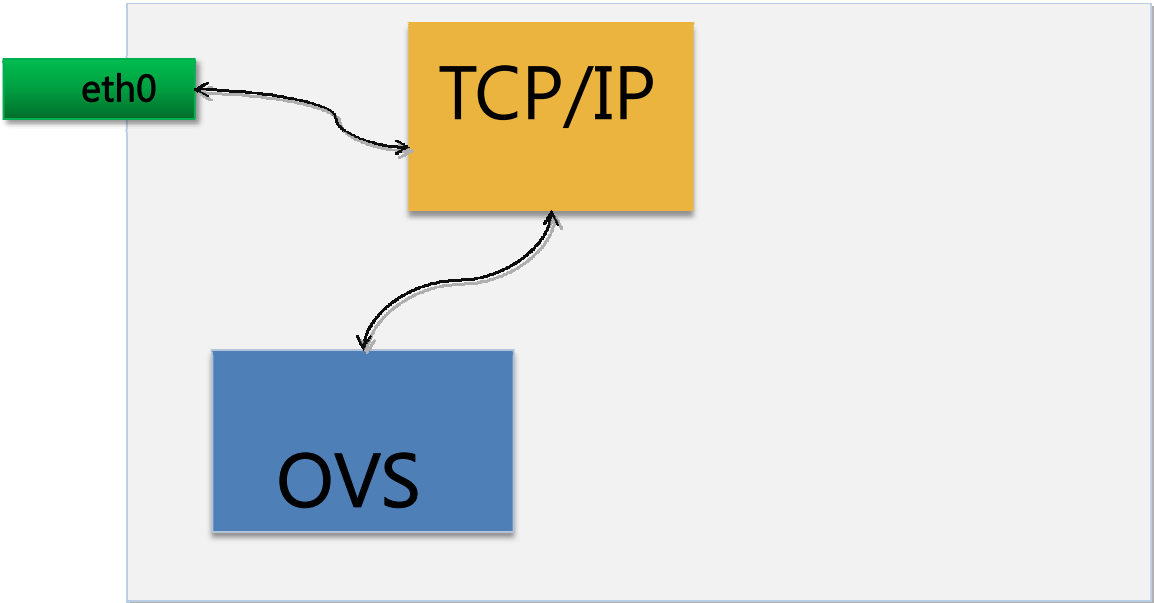
- OVS comprend :
 - **vSwitchd** : le service OVS en mode user
 - **Ovsdb-server** : Contient toutes les configurations d'ovs
 - Un module noyau



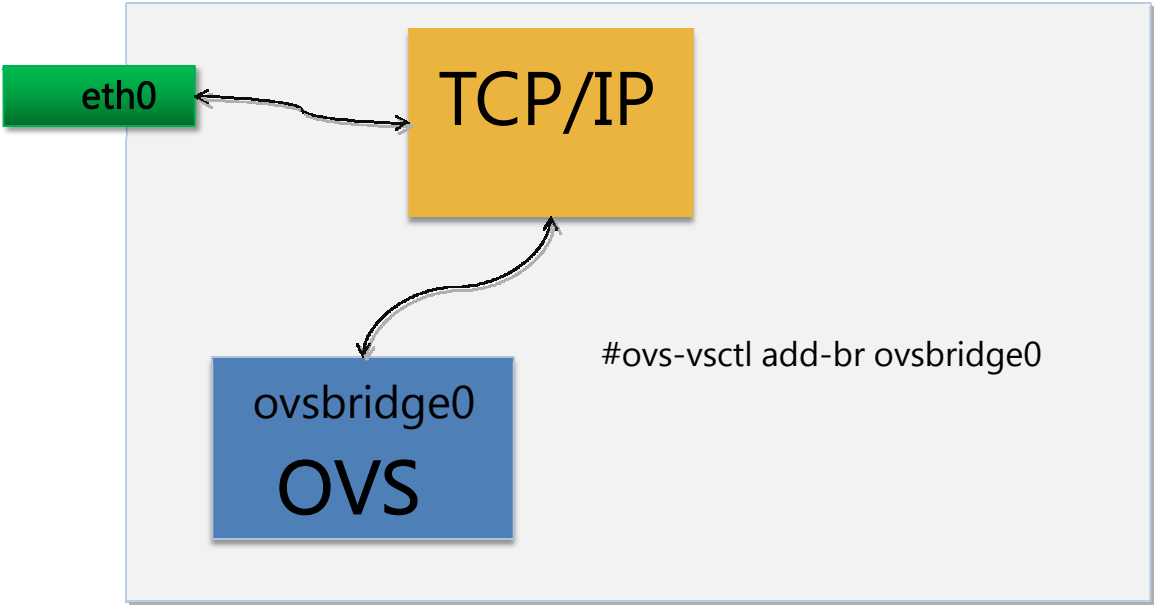
 Configuration basique



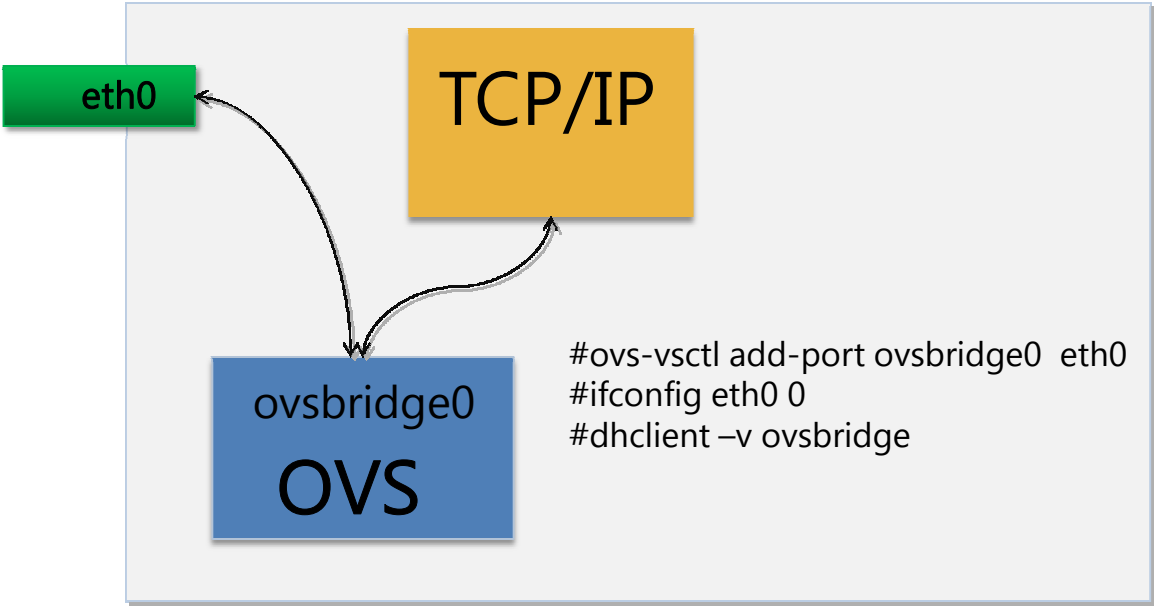
 Configuration basique



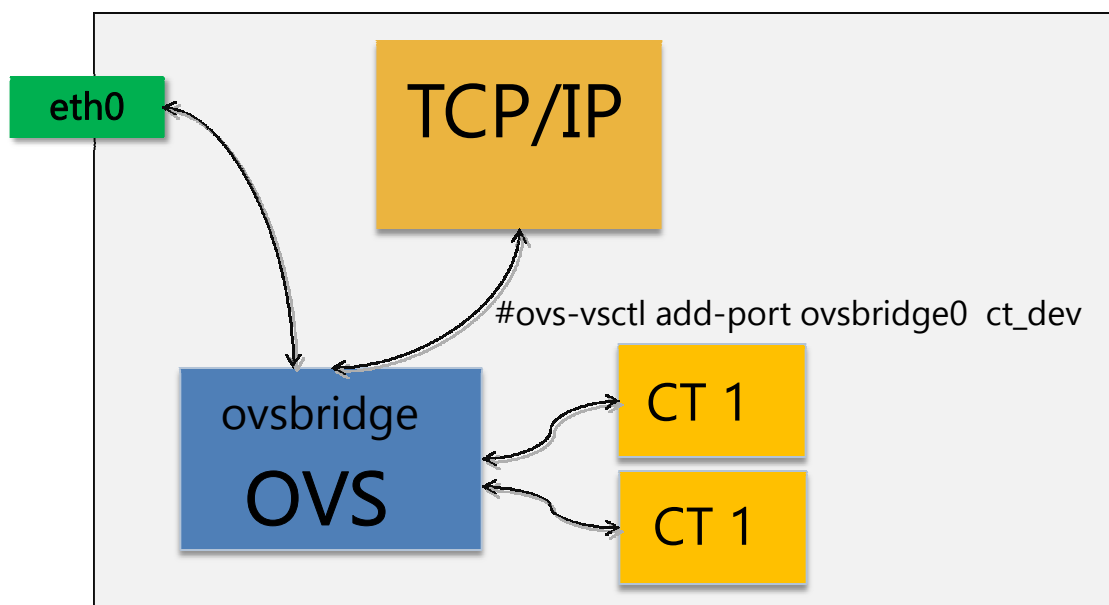
 Configuration basique



 Configuration basique



Configuration basique



Configuration de LXC et des conteneurs

- Installation d'openVswitch
- Création des script d'activation/désactivation des ports ovs
- \$5 est l'interface du conteneur

```
#cat /etc/network/if-up.d/lxc-ifup
#!/bin/bash
ovsBr='lxcbr'
ovs-vsctl --may-exist add-br ${ovsBr}
ovs-vsctl --may-exist add-port ${ovsBr} $5
```

```
#cat /etc/network/if-down.d/lxc-ifdown
#!/bin/bash
ovsBr='lxcbr'
ovs-vsctl --if-exists del-port ${ovsBr} $5
```



Configuration de LXC et des conteneurs

- Configuration des conteneurs

```
#cat /var/lib/lxc/mon_ct_ovs
# Network configuration
lxc.network.type = veth
lxc.network.flags = up
lxc.network.script.up = /etc/network/if-up.d/lxc-ifup
lxc.network.script.down = /etc/network/if-down.d/lxc-ifdown
lxc.network.hwaddr = fe:51:4a:ff:70:7a
lxc.network.veth.pair = port0_mon_ct      #OPTIONNEL
```

- Démarrage du conteneurs

```
# lxc-start -n mon_ct_ovs
```



Ce qu'on a couvert

- Présentation d'OpenvSitch
- Architecture d'OpenvSwitch
- Configuration basique
- Configuration de LXC et des conteneurs





Alphorm.com

Configuration avancée des containers

Les Backends de stockage

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Les Stockage backingstores
- Le stockage par répertoire
- Le stockage LVM
- Le stockage ZFS
- Le stockage BTRFS
- Comparatif ZFS - BTRFS



Formation Virtualisation légère avec LXC

alphorm.com™©



Les Storage backingsstores

- LXC supporte plusieurs variétés de stockage (file system) pour ses conteneurs, appelés **Storage backingsstores**
- Le stockage par répertoire (mode par défaut)
 - `/var/lib/lxc/mon_ct/rootfs`
- L'overlayfs : Utilisé principalement pour les clonages "liés" ou clonage par snapshot
 - `overlayfs:/var/lib/lxc/mon_ct/rootfs:/var/lib/lxc/mon_ct2/delta0`
- Le stockage LVM, un volume logique par conteneur
 - `/dev/lxc/mon_ct`
- Le stockage ZFS, un file system par conteneur
 - `Pool lxc/conteneur/mon_ct`
- Le stockage BTRFS, un sous volume par conteneur
 - `ID 258 gen 20 top level 5 path mon_ct_btrfs_t/rootfs`



Le stockage par répertoire

- Stockage natif sur les systèmes de fichiers standard (ext4, xfs, ..)
- Gestion des snapshots avec overlayfs, hors ligne
- Gestion des clones complets et avec overlayfs
- Pas de répliquations de données
- Pas de protection native (raid, snapshot natif du fs, ..)
- Pas de possibilité d'étendre et réduire le système de fichier

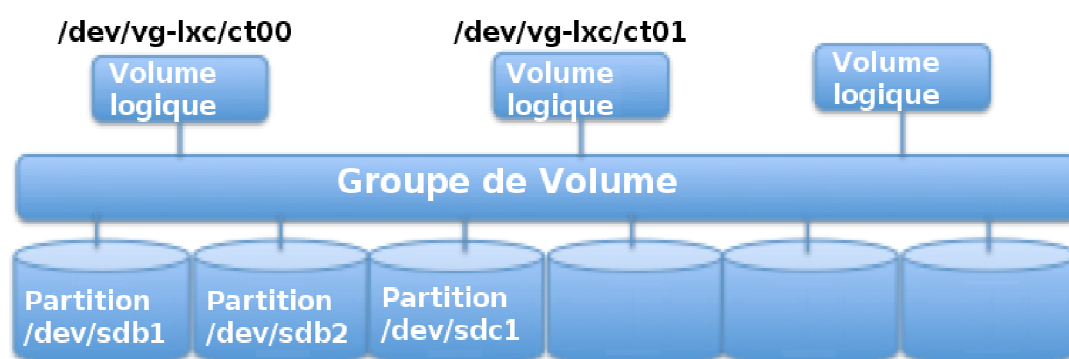


Le stockage LVM

- Supporté depuis longtemps sous Linux
- Gestion des snapshots, clichés à chaud
- Gestion du "mirroring" et agrégés par bandes
- Extensions de l'espace disque à chaud
- Pas de gestion de type RAID
- Simple d'utilisation



Le stockage LVM



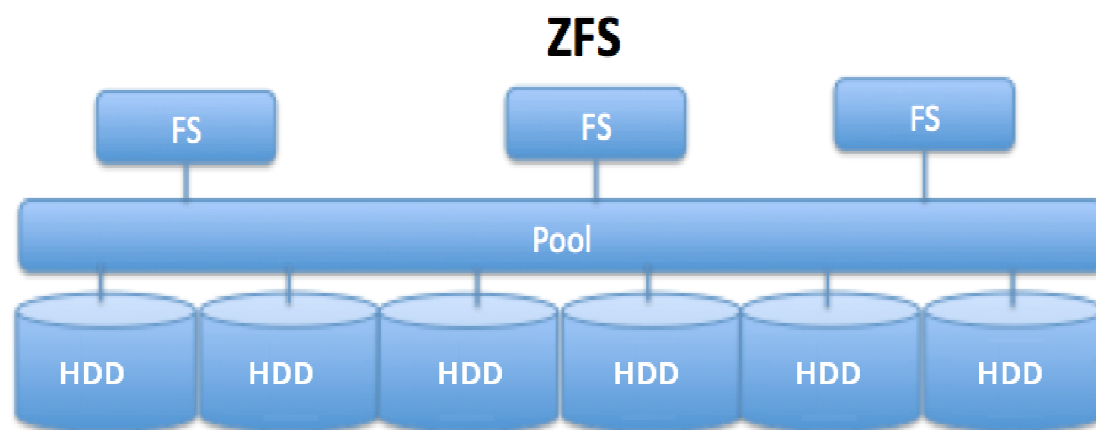
Le stockage ZFS

- GIGANTESQUES capacités de stockage
- Support natif du RAID :
 - RAID-0, RAID-1, RAID-Z (RAID-5 amélioré, parités doubles ou triples)
- Intégrité des données :
 - Les données et métadonnées ont des sommes de contrôle vérifiées à chaque lecture de bloc et mise à jour à chaque écriture
 - « Autoréparation » via les sommes de contrôle (si miroir)
 - Réplication explicite, peut conserver jusqu'à 3 copies des blocs
- Répartition dynamique :
 - Distribue automatiquement les écritures à travers tous les périphériques
 - L'ajout d'un nouveau périphérique n'oblige pas la migration des données
 - Peut allouer les blocs de données sur les disques de façon proportionnelle à leurs performances

Le stockage ZFS

- Compression/décompression à la volée
- Gestion des quotas
- Gestion de réservations d'espaces
- Instantanés et clones
- Envoi/Réception :
 - Permet d'échanger des instantanés pour les sauvegardes sur des systèmes distants
- Déduplication :
 - Permet de détecter des données redondantes et ainsi économiser l'espace disque

Le stockage ZFS



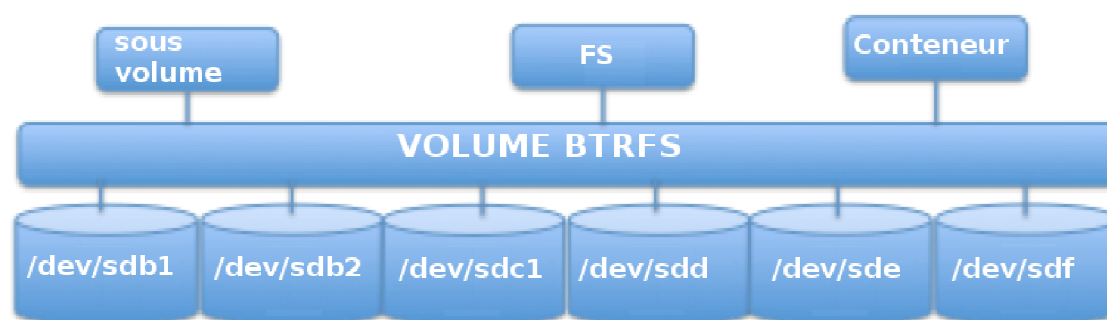
Le stockage BTRFS

- Énormes capacités de stockage
- Plus simple d'utilisation que le ZFS
- Support natif du RAID :
 - RAID-0, RAID-1, RAID-5, RAID-10
- Intégrité des données :
 - Les données et métadonnées ont de sommes de contrôle vérifiées à chaque lecture de bloc et mise à jour à chaque écriture
 - « Autoréparation » via les sommes de contrôle (si miroir)
- Contrôle des périphériques de stockage :
 - Lancement périodique de contrôle des disques en ligne
- Compression/décompression à la volée

Le stockage BTRFS

- Gestion des quotas
- Gestions de réservations d'espaces
- Instantanés et clones, clones "instantanés" (seules les modifications sont stockées)
- Envoi/Réception :
 - Permet d'échanger des clichés pour les sauvegardes sur des systèmes distants
- Déduplication
 - Permet de détecter des données redondantes et ainsi économiser l'espace disque
- Défragmentation

Le stockage BTRFS



Comparatif ZFS - BTRFS

- Performances :
 - BTRFS apparait être plus performant que ZFS sur les systèmes Linux
- Développement :
 - BTRFS est développé par Oracle, Redhat, Fujitsu, Intel, SuSE ET Facebook
 - ZFS a bénéficié du soutien de Sun Microsystems puis Oracle, mais abandonné par Oracle au profit de BTRFS, surtout développé pour FreeBSD
- Licence :
 - BTRFS est distribué sous licence **GPL** (***GNU General Public License***). Son code source est intégré au dépôt officiel du noyau Linux depuis sa version 2.6.29
 - ZFS est quant à lui distribué sous licence **CDDL**. ZFS ne sera jamais directement intégré dans le dépôt officiel des sources de Linux.

Ce qu'on a couvert

- Les Stockage backingstores
- Le stockage par répertoire
- Le stockage LVM
- Le stockage ZFS
- Le stockage BTRFS
- Comparatif ZFS - BTRFS





Alphorm.com

Configuration avancée des containers

LVM et LXC

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Installation et Mise en oeuvre de LVM2
- Création de conteneurs sur les volumes logiques



Formation Virtualisation légère avec LXC

alphorm.com™©



Installation et Mise en oeuvre de LVM2

- **Installation**

```
#apt-get install lvm2
```

- **Création des partitions LVM**

```
#fdisk /dev/sdc
#pvcreate /dev/sdb{1,2,3}
#vgcreate vg-lxc /dev/sdb{1,2,3}
PAS NÉCESSAIRE
#lvcreate -L 5G -n mon_conteneur_on_lvm vg-lxc
```



Création de conteneurs sur les volumes logiques

- **Création d'un conteneur**

```
#lxc-create -B lvm --vgname vg-lxc -n ubuntu_on_lvm -t download
#lvsdisplay
LV Path          /dev/vg-lxc/ubuntu_on_lvm
LV Name          ubuntu_on_lvm
VG Name          vg-lxc
LV Size          1,00 GiB
```

- **Création d'un conteneur sans volumes logiques avec taille et type de fs**

```
#lxc-create -B lvm --vgname vg-lxc --fssize 2G --fstype ext4 -n ubuntu_on_lvm -t\ download
File descriptor 3 (/var/lib/lxc/ubuntu_on_lvm_2G/partial) leaked on lvcreate invocation. Parent PID
5692: lxc-create
Logical volume "ubuntu_on_lvm_2G" create

#lvsdisplay
LV Path          /dev/vg-lxc/ubuntu_on_lvm_2G
LV Size          2,00 GiB
```

Ce qu'on a couvert

- Installation et Mise en oeuvre de LVM2
- Création de conteneurs sur les volumes logiques



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Configuration avancée des conteneurs

ZFS et LXC

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Installation de ZFS
- Mise en oeuvre
- Création de conteneurs sur ZFS
- Snapshot ZFS



Installation de ZFS

- Installation du ppa (dépôt) pour ZFS
`#add-apt-repository ppa:zfs-native/stable`
- Mise à jour des dépôts
`#apt-get update`
- Installation des paquets ZFS
`#apt-get install ubuntu-zfs`



Mise en oeuvre

- Création du pool ZFS avec et sans raid

```
#zpool create lxc /dev/sdb /dev/sdc
#zpool create lxc raidz /dev/sdb /dev/sdc
#zpool list
#zpool status
```
- Création du dataset lxc/containers en déduplication

```
#zfs create -o dedup=on lxc/containers
#zfs list
```
- Mise en place des snapshot

```
#zpool set listsnapshots=on lxc
#zfs set snapdir=visible lxc/containers
```



Création de conteneurs sur ZFS

- Configuration de LXC pour les conteneurs en ZFS

```
#vi /etc/lxc/lxc.conf
lxc.lxcpath = /lxc/containers
lxc.bdev.zfs.root = lxc/containers
```
- Création d'un conteneurs en ZSF

```
#lxc-create -t ubuntu -n mon_ct_zfs -B zfs
#lxc-create -t ubuntu -n mon_ct_zfs
```



Création de conteneurs sur ZFS

- Configuration de LXC pour les conteneurs en ZFS

```
#vi /etc/lxc/lxc.conf
lxc.lxcpath = /lxc/conteneurs
lxc.bdev.zfs.root = lxc/conteneurs
```

- Création d'un conteneurs en ZSF

```
#lxc-create -t ubuntu -n mon_ct_zfs -B zfs
#zfs list
lxc                773M 5.47G 19K /lxc
lxc/conteneurs      767M 5.47G 411M /lxc/conteneurs
lxc/conteneurs/mon_ct_zfs 356M 5.47G 356M /lxc/conteneurs mon_ct_zfs/rootfs
```



Snapshot ZFS

- Snapshot des conteneurs en ZFS

```
#zfs set snapdir=visible lxc/conteneurs/mon_ct_zfs
#zfs snapshot lxc/conteneurs/mon_ct_zfs
#zfs list -t snapshot
NAME                                USED AVAIL REFER MOUNTPOINT
lxc/conteneurs/mon_ct_zfs@snap1    415K  - 356M  -
```

- Restaurer un snapshot

```
# zfs rollback lxc/conteneurs/mon_ct_zfs@snap1
```

Ce qu'on a couvert

- Installation de ZFS
- Mise en oeuvre
- Création de conteneurs sur ZFS
- Snapshot ZFS



Formation Virtualisation légère avec LXC

alphorm.com™©



Alphorm.com

Configuration avancée des containers

BTRFS et LXC

Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Formation Virtualisation légère avec LXC

alphorm.com™©

Plan

- Installation de BTRFS
- Mise en oeuvre
- Création de conteneurs sur BTRFS



Installation de Btrfs

- Installation

```
#apt-get install btrfs-tools
```

Les paquets supplémentaires suivants seront installés :

liblzo2-2

Les NOUVEAUX paquets suivants seront installés :

btrfs-tools liblzo2-2

....

.....

Paramétrage de btrfs-tools (3.12-1) ...

update-initramfs: deferring update (trigger activated)

Traitement déclenché pour libc-bin (2.19-0ubuntu6) ...

Traitement déclenché pour initramfs-tools (0.103ubuntu4.2) ...

update-initramfs: Generating /boot/initrd.img-3.13.0-32-generic



Mise en oeuvre

- Création du volume btrfs

```
#mkfs.btrfs -L lxc -f /dev/sd{b,c,d}
WARNING! - Btrfs v3.12 IS EXPERIMENTAL
WARNING! - see http://btrfs.wiki.kernel.org before using

Turning ON incompat feature 'extref': increased hardlink limit per file to 65536
adding device /dev/sdc id 2
adding device /dev/sdd id 3
fs created label lxc on /dev/sdb
    nodesize 16384 leafsize 16384 sectorsize 4096 size 6.00GiB

#btrfs filesystem show
Label: 'lxc'  uuid: dd880743-263c-41ff-b1bf-295e24084ab3
    Total devices 3 FS bytes used 112.00KiB
    devid   1 size 2.00GiB used 531.94MiB path /dev/sdb
    devid   2 size 2.00GiB used 212.75MiB path /dev/sdc
    devid   3 size 2.00GiB used 519.94MiB path /dev/sdd
```



Création de conteneurs sur btrfs

- Montage du volume btrfs

```
#grep btrfs /etc/mtab
/dev/sdb /var/lib/lxc btrfs rw,relatime,space_cache 0 0
```

- Création du conteneur dans un sous volume btrfs

```
#lxc-create -n mon_ct_btrfs_t -B btrfs -t ubuntu -- -r precise
#btrfs subvolume list /var/lib/lxc/
ID 258 gen 14 top level 5 path mon_ct_btrfs_t/rootfs
```

- Clone du conteneur dans un sous volume btrfs

```
#lxc-clone -o mon_ct_btrfs_t -n mon_ct_btrfs_clone
Created container mon_ct_btrfs_clone as copy of mon_ct_btrfs_t

#btrfs subvolume list /var/lib/lxc/
ID 258 gen 18 top level 5 path mon_ct_btrfs_t/rootfs
ID 259 gen 18 top level 5 path mon_ct_btrfs_clone/rootfs
```

Ce qu'on a couvert

- Installation de BTRFS
- Mise en oeuvre
- Création de conteneurs sur BTRFS



Alphorm.com

Conclusion

Le mot de la fin

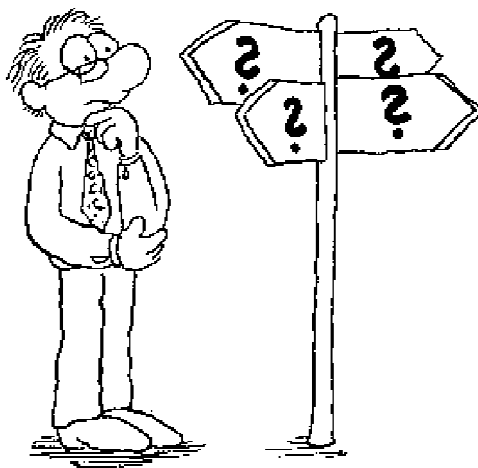
Site : <http://www.alphorm.com>
Blog : <http://blog.alphorm.com>
Forum : <http://forum.alphorm.com>

Ludovic Quenec'hdu
Formateur et Consultant indépendant
OpenSource et virtualisation

Le plan de formation

- Concept
 - La virtualisation par conteneurs
 - Les Namespaces et les Control groups
 - LinuxContainers.org – LXC
- Mise en œuvre
 - Installation et configuration de LXC
 - Containers privilégiés
 - Containers non-privilégiés
- Exploitation des containers avec LXC
 - Commandes de manipulation des conteneurs
 - Le clonage des conteneurs
 - Snapshot
 - Gestion des ressources avec les cgroup
- Exploitation des containers avec LXD
 - LXD présentation et installation
 - Gestion des images LXD
 - Manipulation des conteneurs
 - Les Snapshot
- Exploitation des containers avec LXD
 - Administration des serveurs distants
 - Migration des conteneurs
- Configuration avancée des containers
 - Le modèle réseau Veth, MacVlan, le Bridge et Phys
 - OpenVswitch et LXC
 - Les Backends de stockage
 - LVM et LXC
 - ZFS et LXC
 - BTRFS et LXC

Avez-vous des Questions /Remarques /Commentaires ?





A bientôt 😊

Pour une nouvelle formation sur Alphorm.com

Cursus administration

Red hat 7

