

# J-EOLE 2017

## EOLE Vers une nouvelle gestion des services ?

Gilles Grandgérard  
Gérald Schwartzmann

CC BY-SA 4.0



# Évolution de la gestion des services

Module  $\approx$  rôle prédéfini : Module Amon (2001)

- somme de services prédéfinis
- services activables/désactivable
- ajouts de services possibles de services avec l'installation de paquets supplémentaires

# Évolution de la gestion des services

Module  $\approx$  rôle prédéfini : Démonstrateur utilisant la virtualisation (2007)

- somme de rôles prédéfinis
- utilisation de Xen
- hyperviseur de machine virtuelle
- fait fonctionner plusieurs systèmes d'exploitation virtuels sur une machine hôte

# Évolution de la gestion des services

Module  $\approx$  addition de rôle (pré) défini :

## Module AmonEcole (2009)

- utilisation d'OpenVZ (noyau Linux modifié)
- virtualisation des modules Amon et Scribe

## Module AmonEcole (2011)

- conteneurisation des modules Amon et Scribe
- mode conteneur utilisant la technologie LXC

## Module AmonEcole+ (2012)

- conteneurisation des modules AmonEcole et Eclair
- mode conteneur utilisant la technologie LXC

# Évolution de la gestion des services

Module  $\approx$  autonomisation des services : Eolebase (2013)

- autonomisation : redécoupage de services (empaquetage)
- contextualisation : fonctionnement sur Eolebase et sur d'autres modules
- difficulté de tester tous les contextes d'utilisation

# Évolution de la gestion des services

VM  $\approx$  distribution et addition de module : Modules  
Hâpy (2015)

- solution complète de virtualisation basée sur OpenNebula
- Marketplace OpenNebula pour la distribution

# Apports des solutions EOLE

- Facilité de distribution (paquetage)
- Facilité de déploiement (Zéphir)
- Facilité de configuration (gen\_config)
- Pré-configuration / contexte (gen\_config)
- Authenticité et qualification de la source (dépôts)

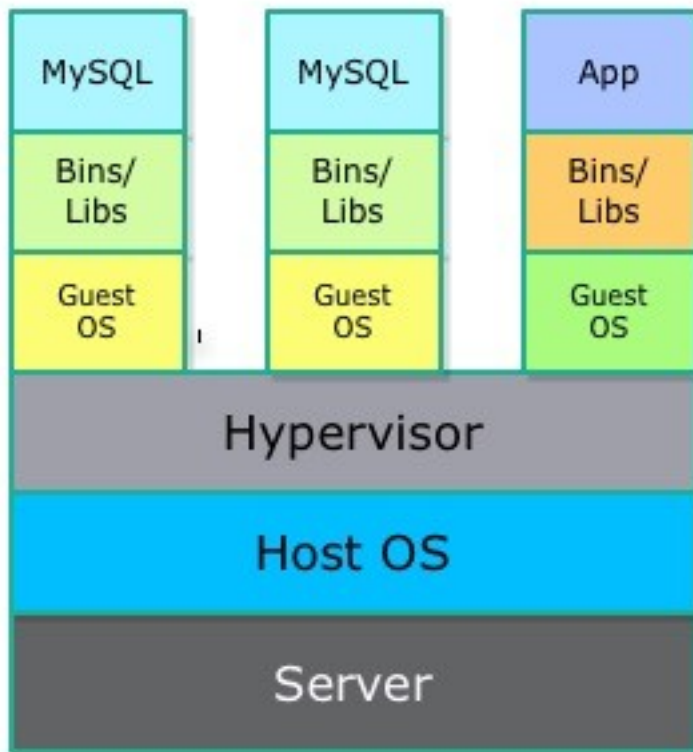
# Vers une nouvelle gestion ?

Les conteneurs...

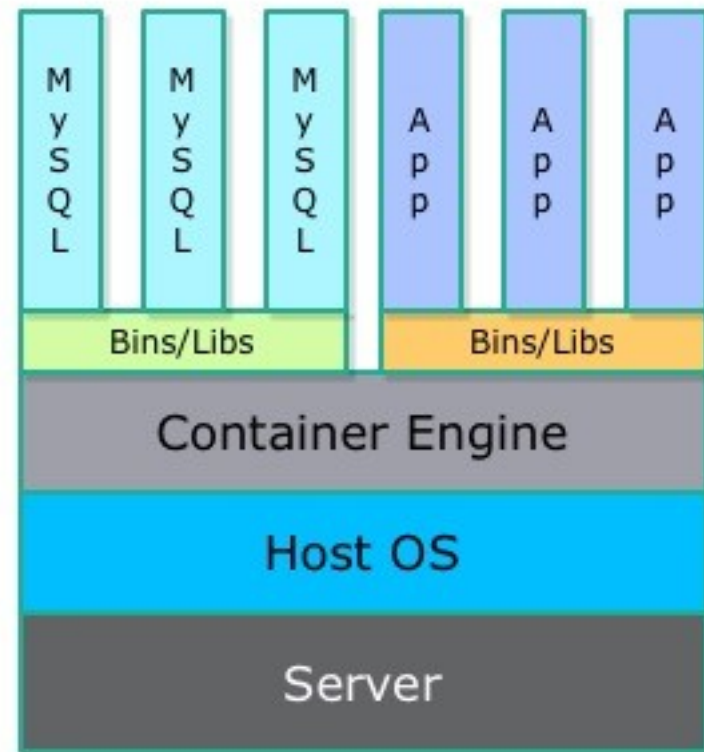


# Vers une nouvelle gestion ?

## Virtual Machines



## Containers



Source de l'image : <http://patg.net/containers,virtualization,docker/2014/06/05/docker-intro/>

# Vers une nouvelle gestion ?

## Avantages

- plus léger que la virtualisation
- s'appuie sur le noyau Linux (espace de nom, cgroups, mécanismes d'isolation additionnels)
- indépendant du système hôte GNU/Linux
- contient l'application et ses dépendances (permet de faire cohabiter plusieurs versions)
- isolation des services (un service  $\approx$  un conteneur)

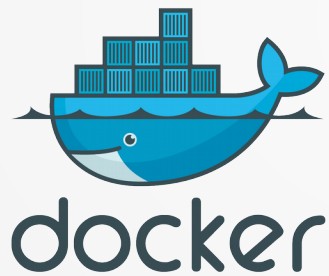
# Vers une nouvelle gestion ?

## Inconvénients

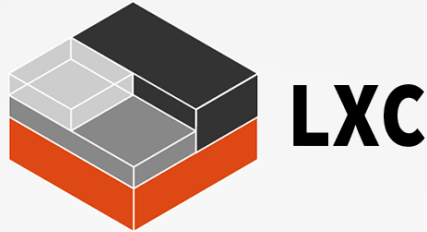
- plus adapté pour des applications de type microservice, plus compliqué pour des applications monolithique
- dépendance plus forte au noyau moins autonome qu'une VM (système d'exploitation limité)
- isolation moins élevée qu'avec un hyperviseur (niveau d'autorisation élevé, partage du noyau)
- rapidité d'installation / rapidité de prolifération
- outils de surveillance et de gestion encore naissants (Kubernetes, DockerUI, Logspout...)

# Vers une nouvelle gestion ?

Quelques technologies de conteneur

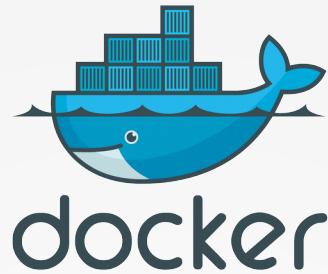


# Vers une nouvelle gestion ?



- virtualise (isole) l'environnement d'exécution
- indépendant du système GNU/Linux
- environnement aussi proche que possible de celui d'une machine virtuelle
- utilise les fonctionnalités du noyau Linux

# Vers une nouvelle gestion ?



- basé sur LXC
- génération automatisée du conteneur (build) qui facilite la distribution
- distribution et gestion de version (sources versionnées avec Git)
- gestion de version sur la machine hôte (instantané et retour en arrière)
- description YAML
- automatisation et rationalisation (Chef, Puppet, SaltStack...)

# Travaux exploratoires

- démonstrateur génération de la configuration Nginx au Hackathon
- tests avec Docker et une registry EOLE
- LDAP (conteneur LXC Alpine) / ejabberd (LXC Ubuntu) OverlayFS snapshot et migration de conteneur déploiement avec SaltStack
- conteneur applications web avec EoleSSO distant
- test de dockeurisation du service EoleSSO

# Démonstration

## Conteneur-2

### Configuration

**B** Nom du conteneur

\* my-apache-app

**N** Technologie du conteneur

\* docker

**N** URL du Hub (Hub Docker par défaut)

**N** Nom du dépôt

httpd

**N** Tag (optionnel)

2.4

**N** Répertoire de l'hôte partagé avec le conteneur

**N** Répertoire de l'hôte partagé avec le conteneur

/var/eole-httpd/

**N** Répertoire du conteneur partagé avec l'hôte

/usr/local/apache2/htdocs/

☰ Montrer/Cacher

+ 📁 Répertoire de l'hôte partagé avec le conteneur

**N** Port du conteneur

**N** Port du conteneur

80

**N** Port de l'hôte

8081

☰ Montrer/Cacher

+ 📁 Port du conteneur

**N** Activer la persistance du conteneur

\* non



# Démonstration

```
root@eolebase:~# cat /etc/nginx/web.d/conteneurs2.conf
location /my-apache-app {
    access_log /var/log/nginx/conteneur2-access.log;
    error_log /var/log/nginx/conteneur2-error.log;
    proxy_pass http://127.0.0.1:8081/;
    proxy_set_header    X-Forwarded-Proto $scheme;

    # ssh_eth0
    allow all;

    # Deny by default
    deny all;
}
root@eolebase:~#
```

# Démonstration

Traitement du conteneur **hello-world**  
(conteneur-1)

Récupération de l'image sur le dépôt : `docker pull hello-world`

Using default tag: latest

latest: Pulling from library/hello-world

Digest:

sha256:c5515758d4c5e1e838e9cd307f6c6a0d620b5e07e6f927b07d05f6d12a1ac8d7

Status: Image is up to date for hello-world:latest

Récupération de l'ID du conteneur

Extinction du conteneur 49674a787355 : `docker stop 49674a787355`

Suppression du conteneur 49674a787355 : `docker rm 49674a787355`

`sysctl -w net.ipv4.ip_forward=1`

`net.ipv4.ip_forward = 1`

Démarrage du service : `service docker start`

Lancement du conteneur : `docker run -d --name hello-world hello-world`

a46c5e4e08473a7456a85d2b0067612fc49f33cb61ed5d5fc773573c17156444

Pas de port de sortie

# Démonstration

Traitement du conteneur **my-apache-app**  
(conteneur-2)

Récupération de l'image sur le dépôt : `docker pull httpd:2.4`

2.4: Pulling from library/httpd

Digest:

sha256:72b55a7c15a4ee3d56ff19f83b57b82287714f91070b1f556a54e90da5eee3fa

Status: Image is up to date for httpd:2.4

Récupération de l'ID du conteneur

Extinction du conteneur 7e8d4ea50300 : `docker stop 7e8d4ea50300`

Image persistante pas de suppression du conteneur

`sysctl -w net.ipv4.ip_forward=1`

`net.ipv4.ip_forward = 1`

Démarrage du service : `service docker start`

Démarrage du conteneur : `docker start my-apache-app`

Numéro de port de sortie à configurer : 8081

Création du fichier de configuration nginx

<https://eolebase.ac-test.fr/my-apache-app/>

# Orientation des travaux

- interaction entre les services
- utilisation de conteneur de données
- personnalisation de la configuration
- gestion de la sauvegarde
- orchestration
- automatisation
- Snappy, Vagrant
- ...

# Licence

Cette présentation est mise à disposition sous licence **Creative Commons by-sa 4.0**

Attribution

Partage dans les mêmes conditions

Vous pouvez obtenir une copie de la licence :

– Par internet :

<https://creativecommons.org/licenses/by-sa/4.0>

– Par courrier postal : Creative Commons, 444 Castro Street, Suite 900 Mountain View, California, 94041, USA