Manuel de déploiement

Projet « Train Commander » Supinfo 2015 – 2016

Sommaire

[1 – Rappel de la topologie du réseau 4](#_Toc453165275)

[2 – Matériel et systèmes d’exploitation requis 5](#_Toc453165276)

[1 – Les hyperviseurs 5](#_Toc453165277)

[2 – Les machines serveur 5](#_Toc453165278)

[3 – Remarque 5](#_Toc453165279)

[3 – Installations des machines et hyperviseurs 6](#_Toc453165280)

[1 – Ordre d’installation 6](#_Toc453165281)

[2 – Les machines physiques 6](#_Toc453165282)

[3 – Les hyperviseurs 6](#_Toc453165283)

[4 – Les machines virtuelles 7](#_Toc453165284)

[4 – Avant de commencer 8](#_Toc453165285)

[5 – Configurations de base 9](#_Toc453165286)

[1 – Les machines virtuelles 9](#_Toc453165287)

[1 – Changer le nom de la machine 9](#_Toc453165288)

[2 – Changer l’adresse IP de la machine 9](#_Toc453165289)

[3 – Changer l’adresse du serveur DNS 10](#_Toc453165290)

[4 – SSH 10](#_Toc453165291)

[2 – Les hyperviseurs 11](#_Toc453165292)

[1 – Changer l’adresse IP et le nom de la machine 11](#_Toc453165293)

[2 – Changer l’adresse du serveur DNS 11](#_Toc453165294)

[6 – DNS et Stockage 12](#_Toc453165295)

[1 – Installation des paquets 12](#_Toc453165296)

[2 – Configuration de Bind9 12](#_Toc453165297)

[3 – Configuration de NFS 12](#_Toc453165298)

[7 – Mail serveur 13](#_Toc453165299)

[1 – Installation des paquets 13](#_Toc453165300)

[2 – Génération des certificats 13](#_Toc453165301)

[3 – Configuration de Postfix 14](#_Toc453165302)

[4 – Configuration de Dovecot 14](#_Toc453165303)

[5 – Configuration de rainloop 15](#_Toc453165304)

[8 – Base de données 17](#_Toc453165305)

[1 – Installation des paquets 17](#_Toc453165306)

[2 – Configuration de MySQL 17](#_Toc453165307)

[9 – API 20](#_Toc453165308)

[1 – Installation des paquets 20](#_Toc453165309)

[2 – Configuration de NFS 20](#_Toc453165310)

[3 – Configuration d’Apache 2 20](#_Toc453165311)

[10 – Web 22](#_Toc453165312)

[1 – Installation des paquets 22](#_Toc453165313)

[2 – Configuration de NFS et Apache 22](#_Toc453165314)

[3 – Configuration de Symfony 22](#_Toc453165315)

[4 – Génération des certificats 22](#_Toc453165316)

[11 – Reverse Proxy 24](#_Toc453165317)

[1 – Installation des paquets 24](#_Toc453165318)

[2 – Configuration de Keepalived 24](#_Toc453165319)

[3 – Configuration de HAProxy 24](#_Toc453165320)

[4 – Inspecter l’état des clusters 25](#_Toc453165321)

[5 – Remarque 25](#_Toc453165322)

[12 – CDN 26](#_Toc453165323)

[1 – Installation des paquets 26](#_Toc453165324)

[2 – Avant de configurer 26](#_Toc453165325)

[3 – Configuration de la VIP, Keepalived et HAProxy 26](#_Toc453165326)

[4 – Configuration de Varnish 26](#_Toc453165327)

[14 – XenCenter 27](#_Toc453165328)

[1 – Connexion à un hyperviseur 27](#_Toc453165329)

[2 – Créer un pool d’hyperviseurs 27](#_Toc453165330)

[3 – Création de machines virtuelles 27](#_Toc453165331)

[4 – Duplication des machines virtuelles 28](#_Toc453165332)

[5 – Accès à la machine virtuelle 28](#_Toc453165333)

[6 – Autres sections 28](#_Toc453165334)

[15 – Annexe 29](#_Toc453165335)

[1 – Commandes basiques 29](#_Toc453165336)

[1 – Edition de fichiers 29](#_Toc453165337)

[2 – Déplacement dans les dossiers 29](#_Toc453165338)

[3 – Création d’un dossier 29](#_Toc453165339)

[4 – Création d’un fichier 29](#_Toc453165340)

[5 – Affichage du contenu d’un dossier 29](#_Toc453165341)

[6 – Manuel 29](#_Toc453165342)

[7 – Elévation de droits 30](#_Toc453165343)

[2 – Glossaire 30](#_Toc453165344)

[1 – Machine Virtuelle ou VM (Virtual Machine) 30](#_Toc453165345)

[2 – Hyperviseur 30](#_Toc453165346)

[3 – Shell 30](#_Toc453165347)

[4 – IP virtuelle ou VIP (Virtual IP) 30](#_Toc453165348)

[5 – Tolérance de panne, Haute disponibilité ou HA (High Availability) 30](#_Toc453165349)

[6 – Distribution de la charge ou load-balancing 31](#_Toc453165350)

[3 – Liens externe 31](#_Toc453165351)

[1 – WinScp 31](#_Toc453165352)

[2 – XenServer 31](#_Toc453165353)

[3 – Citrix 31](#_Toc453165354)

[4 – Debian 31](#_Toc453165355)

[5 – XenCenter 31](#_Toc453165356)

[6 – Bind9 31](#_Toc453165357)

[7 – NFS 31](#_Toc453165358)

[8 – Postfix 31](#_Toc453165359)

[9 – Dovecot 31](#_Toc453165360)

[10 – Rainloop 31](#_Toc453165361)

[11 – Nginx 31](#_Toc453165362)

[12 – MySQL 31](#_Toc453165363)

[13 – Symfony 31](#_Toc453165364)

[14 – HAProxy 31](#_Toc453165365)

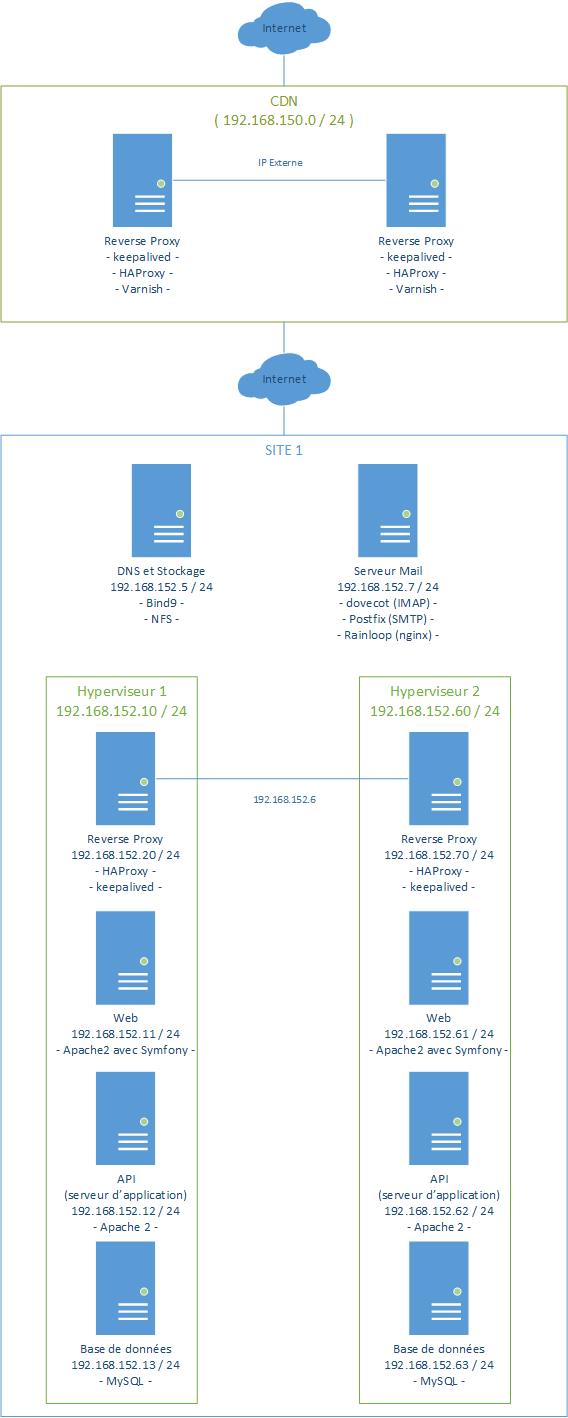
[15 – Keepalived 31](#_Toc453165366)

[16 – Varnish 31](#_Toc453165367)

[17 – SELinux 32](#_Toc453165368)

1 – Rappel de la topologie du réseau

Les instructions de ce manuel utiliseront les adresses IP et la nomenclature utilisée ci-dessous. Veuillez prendre connaissance de la topologie pour mener plus aisément à bien les configurations des serveurs et hyperviseurs.

Notez aussi que la configuration du nom des machines ainsi que leur adresse IP ne seront détaillées qu’une seule fois dans la partie [« Configurations de base ».](#_1_–_Les)

2 – Matériel et systèmes d’exploitation requis

## 1 – Les hyperviseurs

Les [hyperviseurs\*](#_2_–_Hyperviseur) utilisés dans l’architecture sont des [« XenServer »\*](#_1_–_XenServer), de l’entreprise [Citrix\*](#_2_–_Citrix). La version utilisée est la 6.5.0. Il vous est possible de la télécharger depuis le site officiel de Citrix. Il est bon de rappeler que la force de notre architecture est aussi de proposer une architecture gratuite. Ainsi, la version gratuite de « XenServer » est suffisante pour réaliser l’architecture décrite ci-dessous.

Nous avons réalisé ce projet avec seulement 2 Go de RAM, 40 Go de disque dur et un processeur double cœur par hyperviseur. Chaque processeur doit être capable de virtualiser. Il est évident qu’au moins une carte réseau est nécessaire par serveur pour la communication de chaque élément de l’architecture.

## 2 – Les machines serveur

Les machines virtuelles et autres hosts sont toutes des Debian. Elles sont toutes à la version Jessie, soit la version 8, accessible gratuitement via le site officiel de [Debian\*](#_3_–_Debian). Veuillez noter que pour des raisons de performance, nous avons choisi d’utiliser des versions 32 bits pour les machines virtuelles. Il vous sera important de connaître les capacités des processeurs en termes de virtualisation. S’il est vrai que les processeurs capables de virtualiser un OS 64 bits sont nombreux de nos jours, certaines exceptions persistent.

Chaque host virtuel peut se voir attribuer en production 128 Mo de RAM, 8 Go de disque dur et un processeur simple cœur. Toutefois, lors de la configuration de « Composer » pour les serveurs Web, il est nécessaire d’avoir un minimum de 512 Mo. Il vous sera tout à fait possible de repasser ensuite à 128 Mo de RAM.

## 3 – Remarque

Pour les hosts physiques et les serveurs de base de données, il est recommandé d’augmenter le stockage des disques durs pour des raisons de capacités. Les autres besoins matériels sont les mêmes.

La configuration ainsi présentée est faite sous Debian. Néanmoins, il vous est possible de choisir un système Ubuntu, RedHat ou CentOS selon vos affinités car les services, paquets et fonctionnalités y sont présents. Sachez de même que l’entièreté de nos machines serveur n’ont pas d’interface graphique ; libre à vous d’en ajouter une.

3 – Installations des machines et hyperviseurs

## 1 – Ordre d’installation

Pour la bonne configuration et le bon fonctionnement dès l’installation de chacune des machines, il est nécessaire de suivre l’ordre d’installation suivant :

* DNS et stockage
* Server mail
* Hyperviseur 1 et 2
  + Base de données [(VM)\*](#_1_–_Machine)
  + Api (VM)
  + Web (VM)
  + Reverse Proxy (VM)
* CDN

Il n’est pas nécessaire de reproduire l’installation et la configuration des machines virtuelles d’un hyperviseur à l’autre à la main : certaines fonctionnalités de « XenServer » permettent de vous simplifier la tâche. Elles sont décrites dans la partie [« Duplication des machines virtuelles »](#_4_–_Duplication).

## 2 – Les machines physiques

Debian est installable via un CD d’installation, il vous suffit de booter sur celui-ci pour lancer l’installateur. Vous pourrez vous laisser guider à travers les différentes étapes d’installation grâce à l’assistant. Il est conseillé d’avoir une connexion internet lors de l’installation.

Quelques conseils d’installation :

* Choisissez la langue française pour configurer votre clavier en « azerty ».
* Sélectionnez l’entièreté du disque dur puis séparez-le en 2 partitions (une contenant « /home/ »)
* Entrez un nom de machine correcte et un mot de passe complexe pour l’utilisateur « root » (le super-administrateur).
* Lors du choix des logiciels de base, la touche « espace » du clavier coche ou décoche une case alors que « entrée » valide les logiciels sélectionnés sans confirmation.
* Lors de la configuration de « apt » choisissez un miroir français (le premier de la liste est recommandé) pour mettre à jour le système dès l’installation.
* Installez le GRUB sur la partition que vous désirez (celle par défaut convient amplement).

Les adresses IP ne sont pas configurables depuis l’installateur, veuillez-vous en remettre à la partie [« Configurations de base »](#_2_–_Changer) pour plus d’informations.

## 3 – Les hyperviseurs

L’installation des hyperviseurs est plus simple que celle de Debian, car il y a moins d’étapes. Il faut de même booter sur le CD d’installation de XenServer. Veillez à entrer un nom d’host correct et à configurer l’adresse IP en statique avec la valeur définie dans la topologie. Configurez aussi le DNS avec l’adresse de l’host « DNS et Stockage ».

Tout problème, oubli ou erreur lors de l’installation pourra être rattrapé dans la partie [« Configurations de base »](#_2_–_Les).

## 4 – Les machines virtuelles

Pour ajouter une machine virtuelle, le plus simple est de le faire via l’interface [« XenCenter »\*](#_4_–_XenCenter). Ce logiciel est à installer sur un client capable de communiquer avec les hyperviseurs. Il permet de contacter les hyperviseurs et de les gérer à distance à l’aide d’une interface graphique très intuitive. Vous pourrez télécharger gratuitement « XenCenter » sur le site internet de Citrix.

Une fois en votre possession, connectez-vous à un de vos hyperviseurs en tant que « root » en communiquant l’adresse IP.

Cliquez sur « Add New Server » pour ajouter le second hyperviseur.

Cliquez sur « New Pool » et créez un nouveau pool d’hyperviseurs contenant vos deux hyperviseurs. Créer un pool vous permettra de partager les ressources et VM de chacun des hyperviseurs du pool.

Sélectionnez un hyperviseur puis cliquez sur le bouton « New VM » pour ajouter une machine virtuelle sur l’hyperviseur sélectionné.

Laissez-vous guider par l’assistant d’installation pour créer la machine virtuelle.

Une fois la VM créée, double cliquez sur la VM dans la liste des machines pour la lancer, puis dans l’onglet principal cliquez sur l’onglet « Console » pour avoir une vue de la VM et pouvoir interagir avec elle.

A partir de ce point, installez votre système d’exploitation comme sur un serveur physique.

De plus amples informations sur l’utilisation de XenCenter et l’installation de machines virtuelles sont fournies dans la partie [« Duplication des machines virtuelles ».](#_1_–_Connection)

4 – Avant de commencer

La plupart des commandes relatives au système d’exploitation ou aux fonctionnalités basiques d’un ordinateur sont détaillées dans la partie [« Détail des commandes »](#_1_–_Commandes). Vous y trouverez leur nom et utilisation. Ainsi, certaines actions seront justes exposées et non décrites. C’est à vous d’utiliser les commandes correspondantes. Toutes les autres commandes seront décrites pendant la configuration.

La plupart des fichiers de configuration des services ou paquets ne sera, de même, pas détaillée ailleurs qu’en commentaire à l’intérieur de ces fichiers. Seules quelques informations vous seront communiquées, comme par exemple l’utilisation d’une adresse IP spéciale ou un point clef à prendre en compte.

La hiérarchie des dossiers et fichiers de l’archive correspond à celle des serveurs. Configurer un fichier dans « /etc/dovecot/dovecot.conf » reviendra à chercher le fichier « dovecot.conf » enregistré dans le dossier «*<nom-de-la-machine>*/etc/dovecot/ » de l’archive passée avec ce manuel pour le placer dans le dossier correspondant sur la machine. Nous vous invitons vivement à prendre connaissance de ces fichiers pour plus d’informations.

Vous pourrez vous simplifier ce transfert de système de fichier en utilisant des logiciels de transfert de fichier via SSH comme [« WinScp »\*](#_1_–_WinScp).

La configuration de SSH est expliquée dans les [« configurations de base »](#_4_–_SSH).

5 – Configurations de base

## 1 – Les machines virtuelles

La configuration est la même pour toutes les machines Debian et devra être faite pour chaque machine avant leur configuration.

### 1 – Changer le nom de la machine

Lancer les commandes :

|  |
| --- |
| **$ sudo hostname** *<nom-de-machine>*  **$ sudo** *<nom-de-machine>* **> /etc/hostname** |

*<nom-de-machine> est le nouveau nom de votre machine.*

Puis déconnectez-vous et reconnectez-vous ou redémarrez la machine:

|  |
| --- |
| **$ logout**  **$ sudo reboot** |

### 2 – Changer l’adresse IP de la machine

Avec [l’éditeur de texte\*](#_1_–_Edition) de votre choix, modifiez : « /etc/network/interfaces »

|  |  |
| --- | --- |
| **# The loopback network interface**  **auto lo**  **iface lo inet loopback**  **# The primary network interface**  **allow-hotplug eth0**  **iface eth0 inet static**  **address** <adresse-IP>  **netmask** <masque-de-sous-réseau>  **gateway** <passerelle-par-défaut> | # autorise l’activation à chaud  # configuration en statique  # ex : 192.168.152.5  # ex : 255.255.255.0  # ex : 192.168.152.2 |

Puis redémarrez le service réseau, désactivez puis réactivez l’interface ou redémarrez :

|  |
| --- |
| **$ sudo service networking restart**  **$ sudo ifdown eth0 && ifup eth0**  **$ sudo reboot** |

### 3 – Changer l’adresse du serveur DNS

Avec l’éditeur de texte de votre choix, modifiez : « /etc/resolv.conf »

|  |  |
| --- | --- |
| **domain** <nom-du-DNS>  **search** <nom-du-DNS>  **nameserver** <adresse-IP-DNS> | # ex : train-commander.fr  # ex : train-commander.fr  # ex : 192.168.152.5 |

Pour être sûr que le DNS est configuré statiquement, il faut de même s’assurer qu’aucun DHCP local ne puisse fournir une adresse au démarrage de la machine ou au lancement de l’interface. Les adresses IP étant toutes statiques ainsi que les passerelles par défaut, il n’est pas nécessaire de garder le service « DHCP client » actif sur la machine. De ce fait nous allons le désinstaller.

Voici la commande :

|  |
| --- |
| **$ sudo apt-get purge isc-dhcp-client** |

Maintenant votre adresse DNS est configurée statiquement et ne sera plus changée, sauf modification du fichier « /etc/resolv.conf »

### 4 – SSH

Il vous sera important de pouvoir utiliser SSH. Vous pourrez l’utiliser pour importer les fichiers de configuration depuis l’archive du projet. Il n’est pas nécessaire de le configurer, mais autoriser un accès à distance en « root » simplifiera la tâche de configuration.

Installez le paquet nécessaire :

|  |
| --- |
| **$ sudo apt-get install ssh** |

Avec l’éditeur de texte de votre choix, modifiez : « /etc/ssh/sshd\_config »

Cherchez la ligne :

|  |
| --- |
| **[…]**  **PermitRootLogin without-password** |

Remplacez la par :

|  |
| --- |
| **[…]**  **PermitRootLogin yes** |

Redémarrez le service :

|  |
| --- |
| **$ sudo service ssh restart** |

## 2 – Les hyperviseurs

Toutes les configurations se font en physique sur l’hyperviseur.

### 1 – Changer l’adresse IP et le nom de la machine

Dans le menu principal sélectionnez « Network and Management Interface » et cliquez sur « entrée ». Un nouveau menu s’offre à vous, sélectionnez « Configure Management Interface ». Entrez votre mot de passe « root » si demandé. Puis laissez-vous guider par l’assistant de configuration.

### 2 – Changer l’adresse du serveur DNS

Dans le menu principal sélectionnez « Local Command Shell ». Entrez votre mot de passe « root » si demandé.

Avec l’éditeur de texte de votre choix, modifiez : « /etc/resolv.conf »

|  |  |
| --- | --- |
| **nameserver** <adresse-IP-DNS> | # ex : 192.168.152.5 |

Puis déconnectez-vous pour retourner à l’assistant de configuration grâce à la commande:

|  |
| --- |
| **# logout** |

La désinstallation du service DHCP étant impossible sur « XenServer », votre adresse DNS sera de nouveau changée au démarrage. Toutefois, le DNS n’ayant que fort peu d’utilité dans l’utilisation du « XenServer », il n’est pas nécessaire de le configurer statiquement.

6 – DNS et Stockage

## 1 – Installation des paquets

Il y a deux paquets à installer pour ce serveur. [« Bind9 »\*](#_5_–_Bind9) offre un service DNS extrêmement performant et complet, reconnu comme l’un des meilleurs sous Debian. Le service [« NFS »\*](#_6_–_NFS), quant à lui, permet de créer un serveur de stockage à distance.

Installation :

|  |
| --- |
| **$ sudo apt-get install bind9 nfs-kernel-server** |

## 2 – Configuration de Bind9

Nous allons créer deux zones DNS, une zone locale, pour la communication entre les serveurs en local, nommée « train-commander.lan » puis une zone externe pour les clients se connectant au site internet « train-commander.fr ».

La zone locale doit être aussi capable de résoudre les noms de domaine externes, ainsi on le configure pour qu’il fasse résoudre les noms à d’autres DNS.

Copiez les fichiers configurés de l’archive dans « /etc/bind/ ».

Redémarrez le service bind9 :

|  |
| --- |
| **$ sudo service bind9 restart** |

## 3 – Configuration de NFS

NFS est très simple de configuration. Il faut juste lui fournir un dossier dans lequel seront stockées les données mises à disposition aux autres machines. Le stockage va permettre la distribution du site internet à tous les serveurs web, ce qui simplifiera le déploiement de nouvelles fonctionnalités ou mise à jour. Le stockage doit être accessible en lecture et écriture. En effet le site utilisant « Symfony » et « Composer », il est nécessaire que les données puissent être modifiées à distance.

Pour créer de l’espace de stockage, on crée un dossier « /home/git/ ». Le nom n’a que peu d’importance tant que vous renseignez le bon nom dans le fichier de configuration (cf : suite). Ici le choix du dossier résulte de l’utilisation de « git » comme gestionnaire de version. Vous pouvez y insérer le dossier 3PJT de l’archive.

Pour configurer NFS, il vous suffit juste de copier le fichier « /etc/exports » de l’archive dans le dossier « /etc/ » de la machine puis redémarrez le service.

Redémarrez le service NFS :

|  |
| --- |
| **$ sudo service nfs-kernel-server restart** |

7 – Mail serveur

## 1 – Installation des paquets

Plusieurs paquets vont être installés car ce serveur propose de nombreuses fonctionnalités. Tout d’abord, le serveur doit être capable d’envoyer et de recevoir des mails via les protocoles SMTP et IMAP. Ce sera possible par le biais de [« Postfix »\*](#_7_–_Postfix) et [« Dovecot »\*](#_8_–_Dovecot). La réception de mail n’étant pas obligatoire, elle permet de pouvoir tester l’envoi de mails. N’étant pas accessible par un logiciel externe comme Outlook, nous avons aussi ajouté [« Rainloop »\*](#_9_–_Rainloop) qui est une application web de gestion de mail. Ainsi, un serveur web y est aussi installé, c’est le rôle de [« nginx »\*](#_10_–_Nginx).

Installation des paquets de serveur mail:

|  |
| --- |
| **$ sudo apt-get install dovecot-imapd dovecot-pop3d libsasl2-2 libsasl2-modules sasl2-bin dovecot-sieve postfix** |

Installation de Rainloop :

Créez le dossier « /home/rainloop/ » et s’y déplacer puis :

|  |
| --- |
| **$ sudo apt-get install nginx php5-cli php5-curl php5-fpm php5**  **$ sudo wget –qO- http://repository.rainloop.net/installer.php | php** |

Copiez le fichier de configuration de nginx dans « /etc/nginx/sites-available/rainloop » puis:

|  |
| --- |
| **$ sudo ln –s /etc/nginx/sites-available/rainloop /etc/nginx/sites-enabled/** |

Redémarrer nginx :

|  |
| --- |
| **$ sudo service nginx restart** |

## 2 – Génération des certificats

Les certificats vont nous permettre d’utiliser du SMPTS et de l’IMAPS, soit les protocoles sécurisés d’envoi et de réception de mails.

|  |
| --- |
| **$ sudo openssl req -new -x509 -days 3650 -nodes -out "train-commander.fr.cert" -keyout "train-commander.fr.key"** |

*Renseignez les champs demandés successivement jusqu’au bout du processus. Il est important de se souvenir de chaque champ renseigné.*

Créez le certificat « train-commander.fr.pem » puis copiez le tout dans « /etc/ssl/private/ »

|  |
| --- |
| **$ cat train-commander.fr.cert train-commander.fr.key > train-commander.fr.pem**  **$ sudo cp train-commander.fr.\* /etc/ssl/private/** |

Déplacez-vous dans le dossier « /etc/ssl/certs/ » et exécutez :

|  |
| --- |
| **$ sudo openssl req -new -x509 -days 3650 -nodes -out "ca-certificates.crt" -keyout "ca-certificates.key"** |

*Renseignez les mêmes champs que précédemment.*

## 3 – Configuration de Postfix

Postfix va nous servir de serveur SMTP, c’est grâce à lui que l’on sera capable d’envoyer des mails.

Copiez le contenu du dossier « /etc/postfix/ » de l’archive, avec le dossier « sasl/ » dans le dossier correspondant de la machine.

Ne pas redémarrer le service tant que dovecot n’est pas configuré.

La base de données des utilisateurs est gérée par dovecot, postfix gère les alias des utilisateurs.

La dernière étape est de mapper les alias, domaines, mailboxes pour que la configuration soit prise en charge par le service :

|  |
| --- |
| **$ sudo postmap /etc/postfix/vmail\_aliases**  **$ sudo postmap /etc/postfix/vmail\_mailbox**  **$ sudo postmap /etc/postfix/vmail\_domains** |

## 4 – Configuration de Dovecot

Dovecot va nous permettre de recevoir des mails via le protocole IMAP.

Créez un utilisateur et groupe « email » :

|  |
| --- |
| **$ sudo groupadd email -g 7788**  **$ sudo useradd email -r -g 7788 -u 7788 -d /var/email -m -c "mail user"** |

Créez un fichier de log avec les bons droits :

|  |
| --- |
| **$ sudo touch /var/log/dovecot.log**  **$ sudo chown email /var/log/dovecot.log** |

Copiez le contenu de « /etc/dovecot/ » de l’archive sur la machine puis assignez les bons droits aux fichiers de configuration :

|  |
| --- |
| **$ sudo chgrp email /etc/dovecot/dovecot.conf**  **$ sudo chmod g+r /etc/dovecot/dovecot.conf**  **$ sudo chown root:root /etc/dovecot/users.conf**  **$ sudo chmod 755 /etc/dovecot/users.conf** |

La base de données des utilisateurs de dovecot est dans « /etc/dovecot/users.conf » et contient deux utilisateurs « admin » et « noreply » qui ont le hash en SHA1 de « Commander.2016 » en tant que mot de passe. Ce sont les identifiants de connexion.

Redémarrez les services postfix et dovecot :

|  |
| --- |
| **$ sudo service postfix restart && service dovecot restart** |

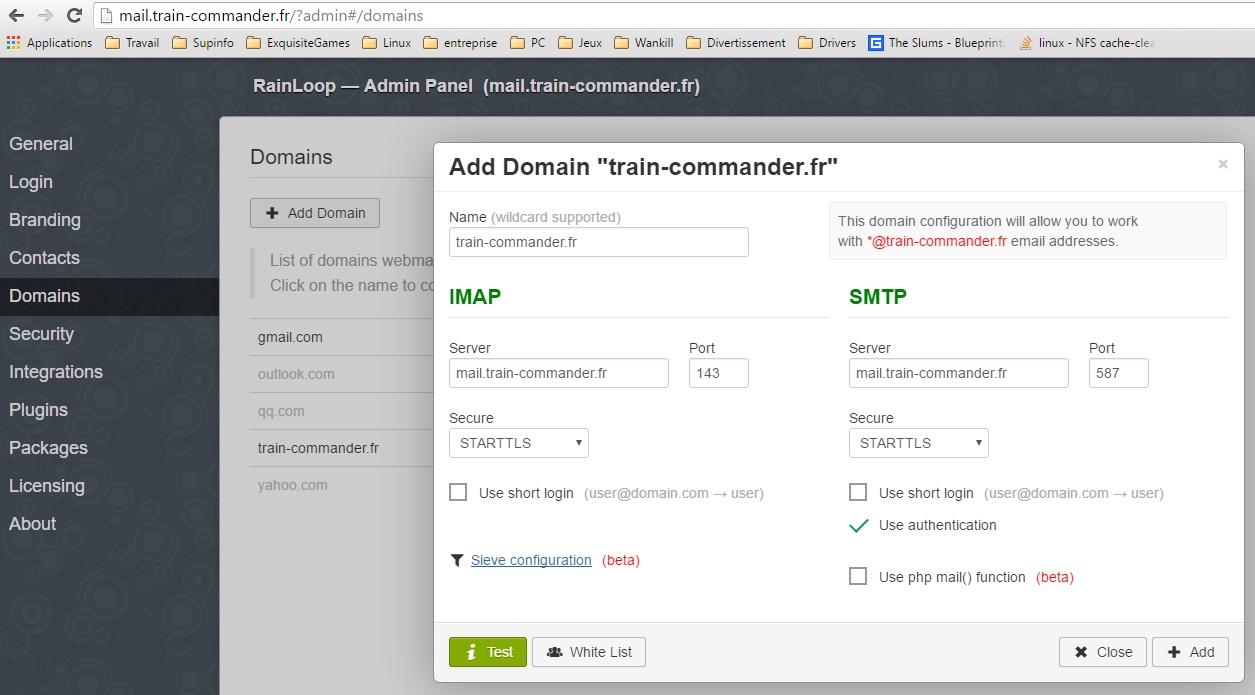
## 5 – Configuration de rainloop

Rainloop va nous permettre de gérer la boite mail des comptes présent grâce à une interface web.

Pour la configurer, connectez-vous à l’aide d’un navigateur sur l’adresse :

*<ip-du-serveur>*/?admin

Le login est « admin » et mot de passe « 12345 ». Dans l’onglet « Domains » cliquez sur « Add Domains » et entrez :



Lors du clic sur « Test » tout doit s’afficher en vert, sinon une erreur a été rencontrée. Cliquez sur « Add » une fois configuré.

Vous pouvez à présent vous connecter sur un des comptes « admin » ou « noreply » sur la page : *<ip-du-serveur>*/

Les identifiants sont : « admin@train-commander.fr » et « noreply@train-commander.fr » et le mot de passe est « Commander.2016 ».

8 – Base de données

## 1 – Installation des paquets

Le serveur de base de données va regrouper toutes les informations relatives aux trains, trajets, utilisateurs et leur historique. Le choix de la base de données s’est porté sur une base de données relationnelle. Le système de base de données [« MySQL »\*](#_11_–_MySQL) convient, de ce fait, parfaitement à l’attente fonctionnelle.

Installez la version serveur de MySQL :

|  |
| --- |
| **$ sudo apt-get install mysql-server** |

*Il vous sera demandé un mot de passe pour l’utilisateur « root », veillez à ne pas l’égarer ! Sans celui-ci, vous ne pourrez pas accéder à la base de données.*

## 2 – Configuration de MySQL

La configuration du cluster MySQL nécessite que deux hosts MySQL soient opérationnels. En effet, le but de ce cluster est de créer deux serveurs MySQL en réplication Master-Master pour qu’ils se partagent leurs nouvelles données réciproquement au fur et à mesure de leur utilisation. Cela garantis une charge réduite lors de leur réplication car ils s’échangent des données plus légères. Pour mettre en place simplement le deuxième host, référez-vous à la partie [« Duplication des machines virtuelles »](#_4_–_Duplication).

**Sur les deux hosts :**

Installez MySQL de manière sécurisée :

|  |
| --- |
| **$ sudo mysql\_secure\_installation** |

*Fournissez votre mot de passe pour l’utilisateur « root » de MySQL puis suivez le guide. Il n’est pas nécessaire de changer le mot de passe « root » mais il est vivement recommandé de répondre « yes » à toutes les autres configurations.*

Copiez le fichier « /etc/mysql/my.cnf » de l’archive dans le répertoire correspondant sur la machine virtuelle puis redémarrez le service MySQL :

|  |
| --- |
| **$ sudo service mysql restart** |

Connectez-vous en local sur le système de base de données :

|  |
| --- |
| **$ mysql –u root -p** |

*Le –u doit précéder le nom de l’utilisateur se connectant, le –p indique qu’un mot de passe doit être entré pour se connecter, saisissez le lorsqu’il vous sera demandé.*

A présent votre [Shell\*](#_3_–_Shell) a changé, il doit ressembler à « mysql> » ce qui signifie que vous êtes connectés. Pour la réplication, il est nécessaire de créer un nouvel utilisateur et lui donner les droits de réplication :

|  |
| --- |
| **mysql> create user 'replicator'@'%' identified by '***<mot-de-passe>***';**  **mysql> grant replication slave on \*.\* to 'replicator'@'%';** |

*<mot-de-passe> est le mot de passe de l’utilisateur « replicator » vous pouvez y mettre ce que vous voulez. Notez que le « ; » en fin de ligne est important, la requête ne se lancera pas tant qu’il ne sera pas renseigné.*

**Sur le premier serveur :**

A présent, nous pouvons voir l’état du nœud en entrant la commande :

|  |
| --- |
| **mysql> show master status;** |

Un tableau similaire à celui-ci devrait apparaitre, les informations ci-présentes seront à utiliser pour le reste de la configuration :

|  |  |  |  |
| --- | --- | --- | --- |
| File | Position | Binlog\_Do\_Db | Binlog\_Ignore\_Db |
| mysql-bin.<nombre> | <position> | traincommanderbdd |  |

**Sur le deuxième serveur :**

Assignez le serveur en tant que paire de réplication avec le premier serveur :

|  |
| --- |
| **mysql> slave stop;**  **mysql> CHANGE MASTER TO MASTER\_HOST = ‘***<IP-du-premier-serveur>***’, MASTER\_USER = ‘replicator’, MASTER\_PASSWORD = ‘***<mot-de-passe-de-replicator>***’, MASTER\_LOG\_FILE = ‘mysql-bin.***<nombre>***’, MASTER\_LOG\_POS =** *<position>***;**  **mysql> slave start;** |

*<IP-du-premier-serveur> : nom de l’hôte du premier serveur sql (sql1.train-commander.lan), <nombre> est le nombre de la colonne « File » du tableau ci-dessus, <position> est le nombre de la colonne « Position » du tableau ci-dessus.*

A présent, nous pouvons voir l’état du nœud en entrant la commande :

|  |
| --- |
| **mysql> show master status;** |

Un tableau similaire à celui-ci devrait apparaitre, les informations ci-présentes seront à utiliser pour le reste de la configuration :

|  |  |  |  |
| --- | --- | --- | --- |
| File | Position | Binlog\_Do\_Db | Binlog\_Ignore\_Db |
| mysql-bin.<nombre> | <position> | traincommanderbdd |  |

**Sur le premier serveur :**

Assignez le serveur en tant que paire de réplication avec le premier serveur :

|  |
| --- |
| **mysql> slave stop;**  **mysql> CHANGE MASTER TO MASTER\_HOST = ‘***<IP-du-deuxieme-serveur>***’, MASTER\_USER = ‘replicator’, MASTER\_PASSWORD = ‘***<mot-de-passe-de-replicator>***’, MASTER\_LOG\_FILE = ‘mysql-bin.***<nombre>***’, MASTER\_LOG\_POS =** *<position>***;**  **mysql> slave start;** |

*<IP-du-premier-serveur> : nom de l’hôte du deuxième serveur sql (sql2.train-commander.lan), <nombre> est le nombre de la colonne « File » du tableau ci-dessus, <position> est le nombre de la colonne « Position » du tableau ci-dessus.*

Le cluster est à présent configuré.

Pour terminer la configuration de MySQL, créez un nouvel utilisateur qui sera utilisé plus tard par le cluster et créez la base de données « trainccommanderbdd ».

**Sur n’importe quel serveur :**

|  |
| --- |
| **mysql> CREATE DATABASE traincommanderbdd;**  **mysql> INSERT INTO mysql.user (Host,User) VALUES ('192.168.152.%','haproxy\_check');**  **mysql> FLUSH PRIVILEGES;** |

9 – API

## 1 – Installation des paquets

Le serveur API est le corps de l’application, il contient l’algorithme de recherche de trajets et accède à la base de données. Il est utilisé par le serveur Web et par l’application mobile. Un serveur de Web API est avant tout un serveur web, ainsi nous allons utiliser « Apache 2 » qui est réputé pour sa stabilité et ses performances. Le code de la web API est entièrement en PHP, il faudra y ajouter les dépendances nécessaires.

Installation d’Apache 2 avec php5 :

|  |
| --- |
| **$ sudo apt-get install apache2 php php5-cli php5-curl php5-mysql libapach2-mod-php5** |

## 2 – Configuration de NFS

Comme exposé précédemment lors de la configuration du serveur « DNS et Stockage », les fichiers du site internet en lui-même sont contenus sur un serveur de stockage, ici un serveur NFS. Nous allons le configurer pour que le serveur web monte automatiquement le serveur NFS au démarrage de la machine.

Tout d’abord, il faut créer le dossier « /mnt/website/ » où sera accessible le stockage distant puis il vous suffira de copier le fichier « /etc/fstab » de l’archive sur la machine. Une fois fait, appliquez les modifications de ce fichier en exécutant la commande :

|  |
| --- |
| **$ sudo mount -a** |

Cela montera tous les systèmes de stockage présent dans le fichier « /etc/fstab » dont le système de stockage distant NFS. Notez que cette commande n’est à taper qu’en cas de modification du fichier « /etc/fstab », le reste du temps, le tout est monté automatiquement au démarrage de la machine.

## 3 – Configuration d’Apache 2

Maintenant que le contenu du site est accessible en local dans « /mnt/website/ », on peut configurer Apache 2 pour qu’il y accède. Mais avant cela, apache doit être capable de faire de modifier des URLs et de modifier les entêtes HTTP. Pour ce faire, il faut ajouter des modules.

Ajout des modules Apache :

|  |
| --- |
| **$ sudo a2enmod rewrite**  **S sudo a2enmod headers** |

A présent, il faut configurer Apache pour qu’il prenne en compte le site internet. Copiez le fichier « /etc/apache2/sites-available/000-default.conf » sur la machine, redémarrez le service Apache 2 :

|  |
| --- |
| **$ sudo service apache2 restart** |

10 – Web

## 1 – Installation des paquets

Le serveur Web est, de même, un simple serveur Apache avec PHP, il a néanmoins la particularité d’utiliser [« Symfony »\*](#_12_–_Symfony) et de gérer le protocole HTTPS.

Vous pouvez vous référer à l’installation du serveur « API » pour l’installation d’Apache et PHP.

Installation de Composer :

|  |
| --- |
| **$ sudo php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"**  **$ sudo php composer-setup.php –install-dir=/usr/local/bin/ --filename=composer** |

## 2 – Configuration de NFS et Apache

Vous pouvez vous référer à la configuration du serveur API pour configurer le serveur. Pensez toutefois à copier les fichiers de configuration du bon serveur !

## 3 – Configuration de Symfony

Pour pouvoir installer Symfony avec Composer, il faut au minimum 512 Mo de RAM lors de la configuration. Symfony utilise du cache et doit être accessible au plus haut niveau de droits.

Créez le dossier de cache et de logs « /var/www/train-commander/cache/ » et « /var/www/train-commander/logs/ » puis donnez-leur les droits maximum :

|  |
| --- |
| **$ sudo chmod 777 /var/www/train-commander/logs/**  **$ sudo chmod 777 /var/www/train-commander/cache/** |

Déplacez-vous dans « /mnt/website/ » et installez les dépendances de Symfony en lançant la commande :

|  |
| --- |
| **/mnt/website/$ composer update** |

A la fin de l’installation, installez les sources du site :

|  |
| --- |
| **/mnt/website/$ php app/console assets:install** |

## 4 – Génération des certificats

Le protocole HTTPS nécessite un certificat pour fonctionner. Vous pouvez récupérer le certificat du serveur mail et l’insérer dans « /etc/ssl/private/traincommander.pem ». Sinon un petit rappel ne fait pas de mal :

|  |
| --- |
| **$ sudo openssl req -new -x509 -days 3650 -nodes -out "traincommander.cert" -keyout "traincommander.key"** |

*Renseignez les champs demandés successivement jusqu’au bout du processus. Il est important de se souvenir de chaque champ renseigné.*

Créez le certificat « train-commander.fr.pem » puis copiez le tout dans « /etc/ssl/private/ »

|  |
| --- |
| **$ cat traincommander.cert traincommander.key > traincommander.pem**  **$ sudo cp traincommander.\* /etc/ssl/private/** |

Redémarrez le service Apache :

|  |
| --- |
| **$ sudo service apache2 restart** |

11 – Reverse Proxy

## 1 – Installation des paquets

Les reverses proxy gèrent le [load-balancing\*](#_6_–_Distribution) de tous les clusters, soit : les serveurs « Web », « API » et « Base de données ». Ils sont tous deux en [haute disponibilité\*](#_5_–_Tolérance) pour gérer toute coupure inopinée ou arrêt de service. Le load-balancing est géré par [« HAProxy »\*](#_13_–_HAProxy) qui offre en plus une interface web permettant de visualiser l’état des différents clusters. La haute disponibilité est sous la responsabilité de [« Keepalived »\*](#_14_–_Keepalived) qui permet de simplement lier une IP virtuelle à une architecture en tolérance de panne.

Installation de HAProxy et Keepalived :

|  |
| --- |
| **$ sudo apt-get install haproxy keepalived** |

## 2 – Configuration de Keepalived

Il est obligatoire de configurer une IP virtuelle, permettant de gérer la tolérance de panne entre les deux serveurs.

Pour cela, importez le fichier « /etc/network/interfaces.vip » sur la machine virtuelle puis entrez cette commande :

|  |
| --- |
| **$ sudo cat /etc/network/interfaces.vip >> /etc/network/interfaces** |

*Le but de cette commande est de copier la totalité du fichier « /etc/network/interfaces.vip » à la fin du fichier « /etc/network/interfaces ».*

Redémarrez le service réseau :

|  |
| --- |
| **$ sudo service networking restart** |

Copiez le fichier « /etc/keepalived/keepalived.conf » sur la machine virtuelle puis redémarrez le service Keepalived :

|  |
| --- |
| **$ sudo service keepalived restart** |

## 3 – Configuration de HAProxy

HAProxy va être configuré pour distribuer la charge entre les serveurs enfant. Ici, on va le configurer pour qu’il gère plusieurs clusters.

Copiez le fichier « /etc/haproxy/haproxy.cfg » sur la machine puis redémarrez le service HAProxy :

|  |
| --- |
| **$ sudo service haproxy restart** |

## 4 – Inspecter l’état des clusters

HAProxy propose une interface web permettant de suivre et voir l’état des serveurs faisant partie des clusters. Comme il est configuré actuellement, il est accessible en local via le site [www.train-commander.lan:8080](http://www.train-commander.lan:8080). L’utilisateur est « admin » et le mot de passe « admin ». Libre à vous de modifier cette configuration.

## 5 – Remarque

Il est tout à fait possible de séparer ce serveur en plusieurs serveurs ne gérant qu’un seul cluster. Pour cela, il suffira de reprendre la même configuration et de changer les informations suivantes :

* La VIP.
* Mettre à jour la Virtual IP dans le fichier de configuration de Keepalived (virtual\_ipaddress).
* Dans le fichier de configuration de Keepalived, la valeur de « virtual\_router\_id » doit être la même pour chaque serveur du cluster, mais différente par cluster pour éviter les collisions entre cluster.
* Ne laisser qu’une seule section « listen » en plus de « defaults », « global » et du « listen adminframe » par cluster dans le fichier de configuration de HAProxy.
* Penser à changer le port de la section « listen adminframe » dans le fichier de configuration de HAProxy de manière à ce qu’il soit différent pour chaque cluster.

Il est possible que vous rencontriez quelques problèmes lors de la configuration de ce serveur, un redémarrage du service Keepalived ou un redémarrage complet de la machine devrait régler ces problèmes.

Redémarrage de Keepalived :

|  |
| --- |
| **$ sudo service keepalived restart** |

Redémarrage de la machine :

|  |
| --- |
| **$ sudo reboot** |

12 – CDN

## 1 – Installation des paquets

L’objectif du CDN est de mettre à disposition le site internet en cache à l’extérieur du réseau local. Le but est que les serveurs du CDN soient plus proches de leurs utilisateurs, ce qui offre un temps de latence plus petit, et donc une meilleure réponse des serveurs pour une expérience utilisateur améliorée. Le CDN est un ensemble de serveurs se distribuant la charge de manière à désencombrer chaque serveur et accélérer les réponses de ceux-ci. La configuration de HAProxy sera ainsi reprise de la section du serveur « Reverse Proxy » pour cette gestion. Nous allons tout de même ajouter un élément clef à ces serveurs : [« Varnish »\*](#_15_–_Varnish), qui a une des solutions de mise en cache HTTP la plus performante à l’heure actuelle.

Reprenez la partie « Installation des paquets » du serveur « Reverse proxy » pour installer HAProxy et Keepalived.

Pour installer Varnish :

|  |
| --- |
| **$ sudo apt-get install varnish** |

## 2 – Avant de configurer

Le réseau complet du CDN étant externe à ceux des sites, les fichiers de configuration ne comporteront aucune adresse IP. Seuls les emplacements des différentes adresses seront indiqués par des balises. Il sera nécessaire pour le bon fonctionnement du CDN que vous les remplaciez à la main dans les fichiers de configuration.

Les balises sont :

* <VIP-CDN> : est l’adresse IP virtuelle des serveurs du CDN.
* <IP-Train-Commander> : est l’adresse IP du site internet train-commander.fr.
* <Serveur-#> (où ‘#’ est un numéro) : est l’adresse IP d’un serveur du CDN.
* <Server> : est l’adresse IP du serveur courant.

## 3 – Configuration de la VIP, Keepalived et HAProxy

Toutes ses configurations sont similaires à celles du serveur « Reverse Proxy », vous pouvez vous en remettre à leur configuration (n’oubliez pas de changer les IP dans les fichiers de configuration).

## 4 – Configuration de Varnish

Copiez simplement le contenu du dossier « /etc/varnish/default.vcl », « /etc/default/varnish », « /etc/systemd/system/multi-user.target.wants/varnish.service » de l’archive sur la machine puis redémarrez tous les services :

|  |
| --- |
| **$ sudo service keepalived restart && service haproxy restart && service varnish restart** |

14 – XenCenter

Nous profiterons de cette section pour expliquer quelques fonctionnalités de base de XenCenter.

## 1 – Connexion à un hyperviseur

Pour ajouter un hyperviseur dans XenCenter, il suffit de vous y connecter.

Pour ce faire, cliquez sur le bouton « Add Server ». Dans le champ « Server : » indiquez l’adresse IP de l’hyperviseur. Laissez le champ « User Name : » pour commencer à « root » puis fournissez le mot de passe de l’utilisateur entré pendant l’installation de l’hyperviseur en question.

Si tout se passe sans accroc, une fois avoir cliqué sur « Add » le serveur apparait dans la liste des serveurs.

## 2 – Créer un pool d’hyperviseurs

Pour créer un pool, il vous faut au moins un hyperviseur actif dans la liste des hyperviseurs. Il est à noter qu’un pool d’un hyperviseur n’ayant pas grande utilité, nous verrons comment ajouter un hyperviseur à un pool déjà existant.

Cliquez sur le bouton « New Pool ». Dans le champ « Name : » renseignez le nom du pool qui sera affiché dans la liste des serveurs. Le champ « Master : » permet de sélectionner l’hyperviseur qui sera en charge du pool. Puis sélectionnez si vous le souhaitez d’autres hyperviseurs à ajouter dans le pool dès sa création.

Pour ajouter un Hyperviseur à un pool déjà existant, clic droit sur cet hyperviseur, dans la liste des serveurs, et cliquez sur « Add to pool… ».

Il arrive que le serveur Master du pool vienne à changer, vous en serez averti et si le changement est fait, l’impact est quasi nul.

## 3 – Création de machines virtuelles

Dans la liste des serveurs, clic droit sur un serveur et cliquez sur « New VM… ».

Le « Template » correspond au système d’exploitation de l’OS, notez que Debian 8 n’est pas présent, mais Debian 7 supporte Debian 8.

Le « Name » Correspond simplement au nom de la machine dans la liste des serveurs et peut être modifié après création en faisant un clic droit sur la machine virtuelle puis « Properties ».

La section « Installation Media » correspond à la manière dont vous allez installer le système d’exploitation. Par défaut, il s’agit d’une installation par CD d’installation à insérer dans le serveur hyperviseur. Toutefois, il est possible d’installer un système d’exploitation via une banque de fichiers ISO dans un stockage distant ou local à l’hyperviseur. Renseignez-vous sur la documentation officielle pour plus d’informations.

La partie « Home Server » permet de sélectionner l’hyperviseur sur lequel la machine sera installée. Notez que seuls les hyperviseurs du même pool que l’hyperviseur courant sont visibles.

Le reste de l’installation correspond aux composants et capacités de votre machine virtuelle, libre à vous de sélectionner ceux et celles qui vous semblent les plus appropriés.

## 4 – Duplication des machines virtuelles

Pour pouvoir dupliquer une machine virtuelle elle doit être éteinte.

Clic droit sur la machine à dupliquer puis cliquez sur « Copy VM… ». Dans le champ « Name : » vous pourrez choisir le nom de la machine une fois dupliquée. Dans « Copy mode » vous pourrez choisir entre « Fast clone » qui fait une copie assez rapide de la machine virtuelle mais qui reste dépendante des ressources de l’ancienne, ou alors vous pouvez choisir « Full clone » qui va créer une copie complète et conforme de la machine sélectionnée à l’exception qu’elle sera totalement indépendante de celle-ci, ce processus est plus long et dépend de la taille du disque à copier.

En choisissant « Full clone » vous aurez de même la possibilité de choisir sur quel serveur du pool d’hyperviseur courant vous pourrez créer la nouvelle VM.

## 5 – Accès à la machine virtuelle

Une fois la machine virtuelle lancée, il faut pouvoir y accéder.

Pour ce faire, sélectionnez la machine à accéder et cliquez sur l’onglet « Console » de la section principale.

## 6 – Autres sections

Il est possible de voir l’état de vos machines virtuelles, hyperviseurs ou pool en vérifiant leurs propriétés ou en inspectant l’onglet « General ».

Les autres onglets offrent de nombreuses autres informations intéressantes, nous ne les décrirons pas toutes. Libre à vous de les découvrir. Nous vous conseillons de même de vous référer à la documentation pour plus de précisions sur chaque option de XenServer et XenCenter.

15 – Annexe

## 1 – Commandes basiques

### 1 – Edition de fichiers

Utilisation de « nano ». Nano propose des fonctionnalités basiques de modification de fichier. Il est comparable au bloc-notes Windows :

|  |
| --- |
| **$ nano** *<chemin-du-fichier>* |

Utilisation de « vi ». Vi est un éditeur de texte plus avancé et requiert un certain temps d’adaptation pour tout utilisateur Windows, il offre par contre un nombre plus important de fonctionnalités que Nano.

|  |
| --- |
| **$ vi** *<chemin-du-fichier>* |

De nombreux autres programmes sont disponibles gratuitement pour l’édition de texte (vim, less, more, etc…), faites-vous votre propre opinion !

### 2 – Déplacement dans les dossiers

Se déplacer dans un dossier est une tâche réalisable grâce à la commande « cd » :

|  |
| --- |
| **$ cd** *<emplacement-du-dossier>* |

### 3 – Création d’un dossier

On peut créer un dossier grâce à la commander « mkdir » :

|  |
| --- |
| **$ mkdir** *<emplacement-du-dossier>* |

### 4 – Création d’un fichier

Pour créer un fichier, on peut simplement lancer le programme d’édition et enregistrer le résultat ou créer un fichier vide grâce à :

|  |
| --- |
| **$ touch** *<emplacement-du-fichier>* |

### 5 – Affichage du contenu d’un dossier

Pour afficher les fichiers et dossiers dans le dossier actuel utilisez « ls » :

|  |
| --- |
| **$ ls -lah** |

### 6 – Manuel

Pour en savoir plus sur une commande et ses options, un fichier ou un dossier, vous pouvez utiliser la commande « man » :

|  |
| --- |
| **$ man** *<emplacement-du-fichier>*  **$ man** *<commande>* |

### 7 – Elévation de droits

Il est assez commun sous linux que vous n’ayez pas les droits suffisants pour lancer une commande, lire ou écrire un fichier ou vous déplacer dans un dossier, vous pouvez contourner ce problème en élevant vos droits le temps d’une commande avec « sudo » :

|  |
| --- |
| **$ sudo** *<commande>* |

## 2 – Glossaire

### 1 – Machine Virtuelle ou VM (Virtual Machine)

Serveur ou client exécuté à l’intérieur d’une machine physique ou d’une autre VM. Il peut y avoir plusieurs VM sur une même machine hôte. Les fonctionnalités d’une VM sont les mêmes qu’une machine physique. Leurs existences non physiques leur permettent d’être manipulées plus aisément qu’une machine physique.

### 2 – Hyperviseur

Serveur capable de contenir un certain nombre de machines virtuelles. Un hyperviseur offre plus de fonctionnalités pour créer, configurer, utiliser et gérer les machines virtuelles car il en fait sa spécialité. Un hyperviseur étant une machine, il peut de même être virtualisé dans un autre hyperviseur.

### 3 – Shell

Nom de l’invité de commande en interface textuelle de Debian. Il est communément composé du nom de l’utilisateur, du nom de la machine et des droits de l’utilisateur. Ce Shell peut être modifié par une commande en cours d’exécution.

### 4 – IP virtuelle ou VIP (Virtual IP)

Adresse IP utilisées dans différents domaines, comme la segmentation d’un réseau en réseaux virtuels ou encore en haute disponibilité ou distribution de la charge.

### 5 – Tolérance de panne, Haute disponibilité ou HA (High Availability)

Procédé impliquant plusieurs machines proposant les mêmes services. Il s’agit aussi d’un cluster de basculement en actif-passif, une des machines est active et propose le service qui lui est attribué jusqu’à ce que le service ne fonctionne plus. Dans ce cas, un autre serveur prend le relais.

### 6 – Distribution de la charge ou load-balancing

Procédé impliquant plusieurs machines proposant les mêmes services. Chaque machine propose le service qui lui est attribué les unes après les autres, se partageant ainsi les demandes de l’utilisateur.

## 3 – Liens externe

### 1 – WinScp

<https://winscp.net/eng/index.php>

### 2 – XenServer

<http://xenserver.org/>

### 3 – Citrix

<https://www.citrix.fr/>

### 4 – Debian

<https://www.debian.org/index.fr.html>

### 5 – XenCenter

<http://xenserver.org/open-source-virtualization-download.html>

### 6 – Bind9

<https://wiki.debian.org/fr/Bind9>

### 7 – NFS

<http://nfs.sourceforge.net/>

### 8 – Postfix

<http://www.postfix.org/>

### 9 – Dovecot

<http://www.dovecot.org/>

### 10 – Rainloop

<http://www.rainloop.net/>

### 11 – Nginx

<https://www.nginx.com/>

### 12 – MySQL

<https://www.mysql.fr/>

### 13 – Symfony

<http://symfony.com/>

### 14 – HAProxy

<http://www.haproxy.org/>

### 15 – Keepalived

<http://www.keepalived.org/>

### 16 – Varnish

<https://varnish-cache.org/>