

## Haute disponibilité d'un service Web dynamique

Propriétés	Description
<b>Type de publication</b>	Côté Labo
<b>Intitulé cour</b>	<b>Haute disponibilité d'un serveur Web dynamique</b>
<b>Intitulé long</b>	<b>Haute disponibilité d'un serveur Web avec réplication de la base de données correspondante.</b>
<b>Module</b>	BTS SIO2 – <b>SISR3 – Exploitation des services</b>
<b>Date de publication</b>	Septembre 2013
<b>Date de modification</b>	Novembre 2018
<b>Version</b>	V2.1
<b>Transversalité</b>	<p><b>SI7 :</b></p> <ul style="list-style-type: none"> <li>Justifier le choix d'une solution de mise en production d'un service</li> <li>Stratégies et techniques associées à la continuité de service</li> <li>Stratégies et techniques de sauvegarde et de restauration de données</li> <li>Stratégies et techniques de répartition et de réplication</li> </ul> <p><b>SISR4 :</b></p> <ul style="list-style-type: none"> <li>Justifier le choix d'une solution de gestion de la disponibilité d'un serveur</li> <li>Installer et configurer une solution de disponibilité de serveurs</li> <li>Disponibilité des systèmes, méthodes, technologies, techniques, normes et standards associés</li> </ul>
<b>Présentation</b>	<p>L'objectif de ce Côté Labo (mis en œuvre en module) est de mettre en place une solution de haute disponibilité pour l'application <i>de gestion de frais</i> du laboratoire pharmaceutique Galaxy-Swiss Bourdin (GSB)</p> <p>Il peut être réalisé à la suite du Côté Labo « Le service Web sécurisé » : <a href="http://www.reseaucerta.org/?q=content/service-web-securise">http://www.reseaucerta.org/?q=content/service-web-securise</a>, mais ce n'est pas obligatoire. La situation de départ nécessite uniquement que l'application Web de gestion de frais de GSB soit installée et opérationnelle : cela peut être aussi l'occasion de vérifier que les étudiants soient capables, en autonomie, de mettre à disposition une application Web accessible via un nom d'hôte pleinement qualifié.</p> <p><b>Ce Côté Labo est scindé en quatre activités :</b></p> <ul style="list-style-type: none"> <li><b>Activité 1 :</b> Installation et première configuration des serveurs primaire (serveur maître) et secondaire (serveur esclave)</li> <li><b>Activité 2 :</b> Configuration des ressources « failover IP » et « serviceWeb »</li> <li><b>Activité 3 :</b> Configuration de la réplication des bases de données</li> <li><b>Activité 4 :</b> Intégration de la solution au cluster</li> </ul>
<b>Activités</b>	<p><b>D1.3 - Mise en production d'un service</b></p> <ul style="list-style-type: none"> <li>A1.3.2 Définition des éléments nécessaires à la continuité d'un service</li> </ul> <p><b>D2.1 - Exploitation des services</b></p> <ul style="list-style-type: none"> <li>A2.1.2 Évaluation et maintien de la qualité de service</li> </ul> <p><b>D3.2 - Installation d'une solution d'infrastructure</b></p> <p><b>D3.3 - Administration et supervision d'une infrastructure</b></p> <ul style="list-style-type: none"> <li>A3.3.1 Administration sur site ou à distance des éléments d'un réseau, de serveurs, de services et d'équipements terminaux</li> </ul>

<b>Pré-requis</b>	Avoir quelques notions sur l'installation, la configuration et l'administration d'un serveur Linux ainsi que sur l'exploitation des services Web et des bases de données (dont sauvegarde et restauration). <b>L'application gestion de frais est installée et opérationnelle. Sur Debian 9, il est nécessaire de mettre en place une version compatible avec php7 (version fournie dans ce Côté Labo) sauf si vous installez php5.</b>
<b>Savoir-faire principaux</b>	<b>En SISR3 :</b> <ul style="list-style-type: none"> <li>• Caractériser les éléments nécessaires à la qualité, à la continuité et à la sécurité d'un service</li> <li>• Installer et configurer les éléments nécessaires à la qualité et à la continuité du service</li> <li>• Valider et documenter la qualité, la continuité et la sécurité d'un service</li> </ul>
<b>Prolongements</b>	<b>En SI7 :</b> <ul style="list-style-type: none"> <li>• Rédiger, mettre en place et tester un plan de continuité d'activité (PCA)</li> </ul> <b>En SISR3 :</b> <ul style="list-style-type: none"> <li>• Assurer la haute disponibilité des autres services présents sur le serveur</li> <li>• Intégrer la répartition de charges</li> </ul> <b>En SISR5 :</b> <ul style="list-style-type: none"> <li>• Superviser le cluster</li> </ul>
<b>Outils</b>	<b>SE :</b> Serveur Linux Debian 9 (stable actuelle) ou ultérieur <b>Serveurs/services :</b> Apache2, PHP7, MariaDB/MySQL-server 5.8 installés et configurés à l'identique sur deux serveurs, Corosync et Pacemaker. <b>Clients :</b> navigateur web sur STA Linux, Windows ou autre système. Outils d'analyse et de tests de bon fonctionnement ainsi que phpMyAdmin. <b>Contexte :</b> organisation/GSB-Organisation.doc.  <b>Site officiel de Pacemaker :</b> <a href="http://clusterlabs.org/">http://clusterlabs.org/</a> <b>Documentation :</b> <a href="http://clusterlabs.org/doc/en-US/Pacemaker/1.1/html/Clusters_from_Scratch">http://clusterlabs.org/doc/en-US/Pacemaker/1.1/html/Clusters_from_Scratch</a>
<b>Mots-clés</b>	Disponibilité, HA, HD, cluster, Heartbeat, Corosync, Pacemaker, réplication.
<b>Durée</b>	12 heures en TP
<b>Auteur(es)</b>	Apollonie Raffalli avec la relecture de Yann Barrot

## La haute disponibilité

La « haute disponibilité » (en anglais « high availability ») regroupe de nombreuses techniques et processus permettant de garantir un certain pourcentage de disponibilité d'un service.

Par exemple, un taux de 99 % de disponibilité assure une disponibilité d'environ 361 jours sur 364 alors qu'un taux de 99,5 % assure une disponibilité de plus de 363 jours sur 365.

La réalité économique fait que les organisations tendent de plus en plus vers des taux encore plus grands comme 99,9 % ou 99,99 % notamment sur certains services critiques. En effet, les conséquences d'une interruption de service sont innombrables et peuvent coûter très cher à tous points de vue.

Par exemple, sur le site <http://www.zdnet.fr>, on pouvait lire qu'une interruption de service de 40 mn le 19 août 2013 aurait fait perdre à Amazon près de 5 millions de dollars.

(<http://www.zdnet.fr/actualites/comme-google-amazon-a-subi-une-panne-informatique-39793254.htm>)

## La haute disponibilité, comment ça marche ?

Pour améliorer la haute disponibilité, il existe de nombreuses solutions correspondant à différents domaines, comme :

- la redondance de services installés sur des serveurs différents et le basculement d'un serveur à l'autre ;
- la redondance des composants des serveurs ou des éléments d'interconnexion ;
- la répartition dynamique des données sur plusieurs disques durs (RAID, NAS, SAN...);
- le stockage des sauvegardes à un emplacement géographique différent ;
- les plans de secours ;
- Etc.

Nous ne traiterons ici que de l'aspect redondance de service.

Dans une configuration très simple que nous allons découvrir, la haute disponibilité nécessite la présence d'un serveur secondaire, fonctionnant sous le même système d'exploitation et fournissant un accès aux services que l'on souhaite rendre « hautement » disponibles.

Ce second serveur configuré à l'identique, en règle générale services arrêtés, surveillera le premier en permanence. En cas de panne du serveur primaire, il la détectera et prendra la relève, devenant alors le nouveau serveur actif.

Selon [http://www-igm.univ-mlv.fr/~dr/XPOSE2006/JEREMIE\\_LEGRAND\\_HAUTE\\_DOSPO/index.htm](http://www-igm.univ-mlv.fr/~dr/XPOSE2006/JEREMIE_LEGRAND_HAUTE_DOSPO/index.htm)

Un des algorithmes utilisés pour ce genre d'opérations est basé sur la tachycardie. Il est appelé « heartbeat », ou battements de cœur. Le serveur actif émet régulièrement des informations sur le réseau pour dire qu'il est vivant, pendant que l'autre écoute passivement.



Si plus aucune information n'arrive au serveur en écoute, celui-ci s'alarme.



Il prend alors le rôle de serveur actif (les services basculent sur ce serveur) et se met à son tour à émettre des battements de cœur.

Si l'autre serveur est restauré, il jouera, au moins dans un premier temps, le rôle de serveur en écoute.



**N'importe quelle machine pourra ainsi tomber en panne sans que l'ensemble ne soit pénalisé.**

Ces techniques seront mises en œuvre via deux outils à installer et configurer :

- **Corosync** qui permet de détecter la défaillance d'un poste grâce à un système de communication et de gérer le cluster en lui-même (on aurait pu tout aussi bien utiliser ici un autre outil comme « Heartbeat »).
- **Pacemaker** qui est un gestionnaire de ressources. Il est chargé de créer, démarrer, arrêter et superviser les ressources du cluster c'est-à-dire les services gérés en cluster et donc inclus dans la continuité de services.

## Contexte

Le laboratoire pharmaceutique Galaxy-Swiss Bourdin (GSB) met à disposition des visiteurs médicaux une application Web sécurisée de gestion des frais de remboursement ([Côté labo « Service Web sécurisé »](#)). Cette application nécessite :

- un serveur Web sécurisé (HTTPS, SSL/TLS) ;
- l'accès à une base de données relationnelle, éventuellement administrable par interface Web.

L'authentification des visiteurs pour l'accès au contenu est gérée par l'application à travers la base de données.

L'entreprise a choisi d'héberger en interne les serveurs exécutant l'application sur un serveur Linux. Par mesure de simplification, les deux serveurs sont sur la même machine physique. À noter aussi que ce Côté Labo peut être réalisé même si le protocole HTTPS n'a pas été mis en œuvre.

**GSB désire disposer d'un serveur de secours prêt à prendre le relais si le serveur principal venait à ne plus être opérationnel.**

Vous disposez, dans la ferme des serveurs, d'une machine virtuelle (et vous êtes en mesure d'en créer d'autres) sur laquelle se trouve l'application Web opérationnelle en HTTP ou HTTPS.

La résolution des noms est prise en charge par un **serveur DNS déjà configuré** qui a (en interne) autorité sur la zone **gsb.coop**.

Votre serveur fait partie de la zone **gsb.coop**, il est accessible par son nom pleinement qualifié déclaré sur le serveur DNS (par exemple, pour les tests, **intralabXX.gsb.coop** où **XX représente les initiales de vos prénoms et noms**).

L'application gestion frais est, quant à elle, accessible via l'URL :

<http://gestionfraisXX.gsb.coop/cAccueil.php> ou <https://gestionfraisXX.gsb.coop/cAccueil.php>

Les activités proposées ont pour objectif de construire, étape par étape, une solution totalement opérationnelle.

**La première activité** consiste à préparer les deux serveurs à l'identique en installant et en procédant à une première configuration des outils de haute disponibilité (fichier *haServiceWeb\_activite1*).

**La deuxième activité** a pour objectif d'intégrer le service Web au cluster composé des 2 serveurs (fichier *haServiceWeb\_activite2*).

**La troisième activité** consiste à configurer la réplication entre les deux serveurs selon une architecture de type maître-esclave. On ne se préoccupe pas ici des problématiques de Corosync et de Pacemaker (fichier *haServiceWeb\_activite3*).

**La quatrième activité** a pour objectif d'obtenir une solution complètement opérationnelle en intégrant le service de base de données dans le cluster (fichier *haServiceWeb\_activite4*).

Les étudiants peuvent travailler seuls ou en binôme.