

**POLYTECHNIQUE  
MONTRÉAL**

LE GÉNIE  
EN PREMIÈRE CLASSE



## INF8480 Systèmes répartis et infonuagique

### TP3 : Initiation aux services de l'infonuagique

**Gabriel Côté-Jones : 1771119**

**Aladin Riabi 1717705**

Polytechnique Montréal

15 Avril 2019

## Tests de performance

```
Terminal
Fichier  Édition  Affichage  Rechercher  Terminal  Aide

Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
Salut Inconnu. Je suis test-server-7w2tjnmo77wa.
61.2178471088
[gacota@l4712-05 deploiemnt sans repartition (master)] $
```

*Fig. 1 : Envoi de 50 requêtes HTTP sans répartition.*

```

Terminal
Fichier  Édition  Affichage  Rechercher  Terminal  Aide

Salut Inconnu. Je suis test2-server1-vozavlaze642.
Salut Inconnu. Je suis test2-server3-ekherqb3pzts.
Salut Inconnu. Je suis test2-server2-muv5ta7abkiz.
Salut Inconnu. Je suis test2-server1-vozavlaze642.
Salut Inconnu. Je suis test2-server3-ekherqb3pzts.
Salut Inconnu. Je suis test2-server2-muv5ta7abkiz.
Salut Inconnu. Je suis test2-server1-vozavlaze642.
Salut Inconnu. Je suis test2-server3-ekherqb3pzts.
Salut Inconnu. Je suis test2-server2-muv5ta7abkiz.
Salut Inconnu. Je suis test2-server1-vozavlaze642.
Salut Inconnu. Je suis test2-server3-ekherqb3pzts.
Salut Inconnu. Je suis test2-server2-muv5ta7abkiz.
Salut Inconnu. Je suis test2-server1-vozavlaze642.
Salut Inconnu. Je suis test2-server3-ekherqb3pzts.
Salut Inconnu. Je suis test2-server2-muv5ta7abkiz.
Salut Inconnu. Je suis test2-server1-vozavlaze642.
Salut Inconnu. Je suis test2-server3-ekherqb3pzts.
Salut Inconnu. Je suis test2-server2-muv5ta7abkiz.
Salut Inconnu. Je suis test2-server1-vozavlaze642.
Salut Inconnu. Je suis test2-server3-ekherqb3pzts.
Salut Inconnu. Je suis test2-server2-muv5ta7abkiz.
Salut Inconnu. Je suis test2-server1-vozavlaze642.
8.53845310211
[qacota@l4712-05 deployment repartition (master)] $

```

Fig. 2 : Envoi de 50 requêtes HTTP avec répartition

Les figures 1 et 2 représentent l'exécution d'un script Python qui envoie 50 requêtes HTTP simultanément à l'adresse IP flottante d'un service sans et avec répartition de charge. Le tableau suivant montre le temps d'exécution en secondes pour chaque cas ainsi que le temps en secondes pour l'envoi séquentiel des requêtes.

	Sans répartition	Avec répartition
<b>Temps de traitement de 50 requêtes HTTP simultanées en secondes</b>	61.2	8.5
<b>Temps de traitement de 50 requêtes HTTP séquentielles en secondes</b>	25	25

Bien qu'il y ait trois fois plus de serveurs pour traiter les requêtes dans le cas de répartition de charge, on voit que le temps de traitement est moins que le tiers du temps de traitement pour le système sans répartition. Ces résultats peuvent être expliqués par le fait que le serveur sans répartition se retrouve dans une situation de surcharge.

Lors de l'envoi des 50 requêtes simultanées au service sans répartition, on voit clairement qu'il y a une surcharge du serveur. Ce dernier cesse périodiquement de traiter les requêtes pendant quelques secondes. Lors de certaines exécutions du script, il cesse complètement de traiter toutes requêtes et envoie un message d'erreur 503 pour les requêtes restantes après 109 secondes.

On peut également voir qu'il y a une surcharge du serveur sans répartition lors de l'envoi de 50 requêtes simultanées en comparant avec l'envoi des requêtes séquentiellement. Cette dernière méthode nous donne un temps total de traitement de 25 secondes pour 50 requêtes. Moins de la moitié du temps de traitement des 50 requêtes simultanées.

## Questions

### Question 1 :

- **Nova :** Nova permet l'approvisionnement de serveurs virtuels. Les composants Nova sont faits pour s'exécuter sur des systèmes Linux. Ils roulent sur ces systèmes en Daemon. C'est-à-dire, ils s'exécutent en arrière-plan et servent à répondre aux requêtes du réseau et sont accessible via API REST. La composante Nova est implémentée en Python.
- **Heat :** Heat est le service d'orchestration d'OpenStack. Ce service permet d'exécuter plusieurs composants/applications sur le cloud à l'aide d'un gabarit. Les gabarits Heat décrivent l'infrastructure du système et peuvent être rédigés à la main dans un fichier texte. Heat s'occupe de la gestion du système pour toute sa durée de vie. De plus, Heat est compatible avec plusieurs formats de gabarit.
- **Neutron :** Neutron est la composante OpenStack qui offre des services de réseautage. Cette composante donne aux utilisateurs d'OpenStack un API leur permettant de créer

des topologies de réseautage dans le cloud ainsi que d'y appliquer les politiques et les configurations de leurs choix. Cela permet aux propriétaires d'avoir un certain contrôle sur le réseau et son trafic pour leur système OpenStack.

- **Swift** : Cette composante offre une API pour la gestion de stockage de données sur le cloud. Swift est fait pour stocker des données lorsqu'on est incertain des requis d'espace ou de mises à l'échelle possibles dans le futur. Cette composante offre aux utilisateurs d'OpenStack un service de stockage bâti pour la mise à l'échelle. Les objets et fichiers sont répartis sur plusieurs disques durs à travers le cloud. La mise à l'échelle se fait horizontalement, c'est-à-dire en ajoutant des serveurs.  
Exemple : Avec l'objet OS::Heat::DeployedServer, il est possible de spécifier le contenant Swift qui servira au stockage des données du serveur.
- **Keystone** : Cette composante offre une API pour l'authentification et la gestion d'utilisateurs. Cette composante supporte plusieurs types d'authentification et est compatible avec d'autres services d'authentification comme LDAP. De plus cette composante permet aux utilisateurs d'obtenir une liste de tous les services disponibles sur le système et de choisir celui qu'ils veulent accéder.  
Exemple : L'objet OS::Keystone::Endpoint permet de lier une URL à un service OpenStack et de spécifier s'il est accessible au public ou restreint aux administrateurs du système.
- **Horizon** : Horizon offre aux utilisateurs et au propriétaire du système une interface graphique pour les autres services d'OpenStack.  
Exemple : Un utilisateur pourrait utiliser un « Dashboard » Horizon plutôt qu'une API REST pour accéder aux services OpenStack.

#### Question 2 :

- **OS::Heat::ResourceGroup** : Permet l'instanciation/exécution d'un certain nombre de serveurs web avec possibilité de mise à l'échelle statique.
- **OS::Neutron::LBaaS::HealthMonitor** : Permet la surveillance de serveurs web et l'avertissement des serveurs/noeuds non disponibles à l'objet qui gère le sous-réseau.
- **OS::Neutron::LBaaS::Pool** : L'objet responsable de la gestion du sous-réseau de serveurs virtuels.
- **OS::Neutron::LBaaS::LoadBalancer** : Distribut les requêtes entre les différents serveurs du sous-réseau (pool).
- **OS::Nova::Server** : La ressource qui s'occupe de gérer un serveur virtuel.

#### Question 3 :

1. **OS::Heat::AutoScalingGroup**
2. **OS::Ceilometer::Alarm** et **OS::Aodh::Alarm**: *alarm\_actions* définit l'action à poser, *threshold* définit le pourcentage d'utilisation du CPU à utiliser comme condition,

*comparison\_operator* définit l'opérateur de comparaison avec la valeur *threshold*, *period* définit le temps d'évaluation.

## Références

1. Documentation OpenStack tirée de : <https://docs.OpenStack.org>
2. Page Wikipédia des composantes OpenStack :  
[https://en.wikipedia.org/wiki/OpenStack#Compute\\_\(Nova\)](https://en.wikipedia.org/wiki/OpenStack#Compute_(Nova))