
Étape 1 : Préparer l'environnement

1. Configurer les serveurs backend

- Assurez-vous d'avoir au moins deux serveurs backend prêts (par exemple **backend1** et **backend2**).
- Ces serveurs doivent héberger une application web accessible sur une adresse IP et un port spécifiques.
- Exemple avec un serveur HTTP simple :

```
# Sur backend1
python3 -m http.server 8080
```

```
# Sur backend2
python3 -m http.server 8080
```

2. Configurer le serveur Apache

- Installez Apache sur un serveur dédié qui servira de load balancer.

```
sudo apt update
sudo apt install apache2
```

3. Activer les modules nécessaires

- Activez les modules requis pour la répartition de charge et l'affinité de session :

```
sudo a2enmod proxy
sudo a2enmod proxy_http
sudo a2enmod proxy_balancer
sudo a2enmod lbmethod_byrequests
sudo systemctl restart apache2
```

Étape 2 : Configurer le fichier de configuration Apache

1. Créer un fichier de configuration pour le load balancer

- Créez un fichier de configuration dédié (exemple : **/etc/apache2/sites-available/load-balancer.conf**).

```
sudo nano /etc/apache2/sites-available/load-balancer.conf
```

2. Ajouter la configuration de base

Ajoutez cette configuration pour mettre en place le load balancer :

```
<Proxy "balancer://mycluster">
    BalancerMember http://192.168.1.101:8080 route=node1
    BalancerMember http://192.168.1.102:8080 route=node2
    ProxySet lbmethod=byrequests stickysession=JSESSIONID
</Proxy>

<VirtualHost *:80>
    ServerName example.com

    ProxyPreserveHost On
    ProxyPass / balancer://mycluster/
    ProxyPassReverse / balancer://mycluster/

    ErrorLog ${APACHE_LOG_DIR}/load_balancer_error.log
    CustomLog ${APACHE_LOG_DIR}/load_balancer_access.log combined
</VirtualHost>
```

- Remplacez **192.168.1.101** et **192.168.1.102** par les adresses IP de vos serveurs backend.

3. Activer le site

- Activez le fichier de configuration :

```
sudo a2ensite load-balancer
sudo systemctl reload apache2
```

Étape 3 : Activer l'outil de gestion du load balancer

1. Ajouter la configuration de **balancer-manager**

Modifiez le fichier pour inclure un accès au tableau de bord de gestion :

```
<Location "/balancer-manager">
    SetHandler balancer-manager
    Require ip 192.168.1.0/24
</Location>
```

2. Redémarrer Apache

- Appliquez les changements :

```
sudo systemctl reload apache2
```

3. Accéder au gestionnaire

- Rendez-vous à <http://example.com/balancer-manager> pour superviser et ajuster vos backends.

Étape 4 : Tester le Load Balancer

1. Tester la répartition de charge

- Envoyez plusieurs requêtes à l'URL du load balancer (par exemple, <http://example.com>) :

```
curl http://example.com
```

- Vérifiez que les réponses proviennent alternativement des serveurs backend.

2. Tester l'affinité de session

- Ajoutez un cookie de session à vos requêtes :

```
curl --cookie "JSESSIONID=12345" http://example.com
```

- Assurez-vous que les requêtes avec le même cookie sont redirigées vers le même backend.

Étape 5 : Surveiller et Debugger

1. Vérifiez les logs

- Consultez les logs Apache pour diagnostiquer tout problème :

```
tail -f /var/log/apache2/load_balancer_error.log
```

2. Surveiller le tableau de bord

- Utilisez [balancer-manager](#) pour voir l'état des serveurs backend et leurs charges.

3. Augmenter la verbosité des logs

- Si nécessaire, activez des logs détaillés pour le module proxy :

```
LogLevel proxy:trace2
```

Étape 6 : Optimiser pour la production

1. Activer HTTPS

- Ajoutez un certificat SSL pour sécuriser les requêtes utilisateur :

```
sudo apt install certbot python3-certbot-apache
sudo certbot --apache
```

2. Configurer des checks de santé

- Ajoutez un mécanisme pour surveiller la santé des serveurs backend (module `mod_proxy_hcheck`).

3. Centraliser les sessions

- Utilisez Redis ou Memcached pour centraliser les sessions et éliminer la dépendance à l'affinité stricte.

Résumé des commandes essentielles

```
# 1. Installer Apache
sudo apt update
sudo apt install apache2

# 2. Activer les modules
sudo a2enmod proxy proxy_http proxy_balancer lbmethod_byrequests

# 3. Créer et activer la configuration du load balancer
sudo nano /etc/apache2/sites-available/load-balancer.conf
sudo a2ensite load-balancer
sudo systemctl reload apache2

# 4. Tester la configuration
curl http://example.com
curl --cookie "JSESSIONID=12345" http://example.com

# 5. Debugger et superviser
tail -f /var/log/apache2/load_balancer_error.log
```

