

# BROTLI sur Debian Stretch

- Objet : Installation pour Apache d'un nouvel algorithme de compression de données qui serait plus efficace de 26 % par rapport aux solutions existantes comme GZIP
- Niveau requis : [avisé](#)
- Commentaires : Ayant créé ce tutoriel sur le forum ( j'aime pas les meta-balises de wiki et son formatage ) ayez le reflex d'y passer.  
Toutefois, je vais "tenter" de transposer son contenu ici : <https://debian-facile.org/viewtopic.php?id=21684>
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Création par [root@rkn](#) 19/06/2018
- Testé par <...> le <...> [Fix Me!](#)
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#) <sup>1)</sup>

Qu'est-ce que BROTLI ?

Des chercheurs de Google viennent de présenter un nouvel algorithme de compression de données qui serait plus efficace de 26 % par rapport aux solutions existantes comme GZIP. Baptisée « Brotli », il s'agit d'une nouvelle méthode qui s'appuie sur une variété de techniques de traitement de données comme le « modelage contextuel de second ordre » ou la « réutilisation de codes entropiques ».

*On va rester simple toutefois...* 😊

Pour faire simple, la compression de données s'applique à de nombreux domaines informatiques, que ce soit pour :

- le stockage de données,
- le traitement de contenu multimédia ou
- les flux web !!!

Ce dernier cas m'intéresse tout particulièrement.

Cela permet une nette accélération de mes sites, surtout pour les terminaux mobiles.



Brotli - qui est totalement open source - ne se distingue pas seulement par sa performance, il introduit également un nouveau format de données.

Actuellement, le format le plus utilisé est « Deflate ». Il est utilisé en particulier pour les fichiers de type gzip ou PNG.



Cet algorithme pourrait être implémenté dans nombre d'applications.

Ce n'est pas la première fois que Google s'intéresse à la compression de données. Il y a deux ans, ses chercheurs ont publié un algorithme open source basé sur le format Deflate et baptisé « Zopfli ». Là encore, le nom s'inspire d'une variété de pain en Suisse allemande.



Une chose importante à noter est que les navigateurs ne supporteront que Brotli via HTTPS.

En effet, certains anciens navigateurs (mandataires ) ne gèrent pas bien les nouveaux formats permettant l'activation des nouvelles fonctionnalités.

Aujourd'hui, tous les navigateurs sont supposés supporter Brotli.

Je ne rentrerai pas dans le détail des Cyphers SSL de nos serveurs, mais, suivant la paranoïa de l'admin, certains OS et certains mandataires peuvent être simplement rejetés a la connection ...

**j'écrirais un petit quelque chose la dessus bientôt.**

## Prérequis:

1. une distro Debian sous Apache ( j'ai temporairement laissé tomber NGINX )
2. le protocole HTTPS fonctionnel.
3. les binaires de Brotli installés.

## Installation:

1 installation des binaires :

```
sudo apt-get install brotli
```

2 installation des packages Apache2-dev

```
sudo apt-get install apache2-dev
```

3 récupération des sources pour l'apache-mode-brotli via git

```
cd /opt/<code user>sudo apt-get install git<code user>sudo git clone --depth=1 --recursive https://github.com/kjdev/apache-mod-brotli.git
```

4 Construction du module apache-mde-brotli :

```
cd /opt/apache-mod-brotli
./autogen.sh
./configure
make
install -D .libs/mod_brotli.so /usr/lib/apache2/modules/mod_brotli.so -m 644
```

5 Initialisation du module :

création du liens logique pour le module

```
cd /etc/apache2/mods-available
echo "LoadModule brotli_module /usr/lib/apache2/modules/mod_brotli.so" >
brotli.load
```

6 Création, tuning de brotli.conf, et démarrage du module :

```
cd /etc/brotli
```

```
nano brotli.conf
```

Exemple de ma config :

```
<IfModule brotli_module>
# Compression
## BrotliCompressionLevel: 0-11 (default: 11)
BrotliCompressionLevel 10
## BrotliWindowSize: 10-24 (default: 22)
BrotliWindowSize 22
# Logging
# Filter note
BrotliFilterNote Input brotli_in
BrotliFilterNote Output brotli_out
BrotliFilterNote Ratio brotli_ratio
#LogFormat '"%r" %{brotli_out}n/%{brotli_in}n (%{brotli_ratio}n)' brotli
#CustomLog ${APACHE_LOG_DIR}/brotli_access.log brotli
# Output filter
AddOutputFilterByType BROTLI text/html text/plain text/css text/xml
AddOutputFilterByType BROTLI text/css
AddOutputFilterByType BROTLI application/x-javascript
application/javascript
AddOutputFilterByType BROTLI application/rss+xml
AddOutputFilterByType BROTLI application/xml
AddOutputFilterByType BROTLI application/json
# Global output filter if necessary.
# SetOutputFilter BROTLI
# SetEnvIfNoCase Request_URI \.txt$ no-br
</IfModule>
# EOF
```

7 Le tuning de votre système dépendant de votre quantités de rem, de votre bande passante et autre joyeusetés, vous pouvez soit garder ma configuration, soit rechercher sur internet.

8 lorsque votre tuning est fini, copy du .conf en .local pour éviter écrasement si mise a jour des binaires dans le futur :

```
cp brotli.conf brotli.local
```

9 démarrage du module :

```
a2enmod brotli
```

```
service apache2 restart
```

10 si besoin, arrêt du module :

```
a2dismod brotli
```

```
service apache2 restart
```

Vous pouvez tester brotli via [les outils de CDN.COM](#)



Et voila.

Pour tester les bénéfices de brotli sur votre serveur, faites un test avant puis un test après avec les outils [de chez CDN.COM](#)

Sur une de mes machines, je suis passé de 45 requetes et 3.7 mb de transfert à 49 requetes et 2.8 mb.

Un exemple :



## Remarques

Cette manipulation est nécessaire a chaque grosse mise a jour d'apache.  
Par exemple de Apache2 2.4.25 à apache 2.4.33 !

.

1)

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc:systeme:brotli>

Last update: **07/09/2019 19:16**

