How to enable HTTP gzip Compression on CentOS 5.x to speed up Apache Webserver

Author : admin

Itâ€™s a wide known fact that the so called **HTTP Compression** provided by mod_gzip module in Apache 1.x and by the mod_deflate module on Apache 2.x saves up a lot of internet traffic by compressing the transferred data in between the client -> server interaction.
Nowadays almost 98% or 99% of the browsers on the net perfectly support the mod_gzip file compression.
Therefore enabling the **http gzip compression** is a must have module in every serious webserver out there which is targetting better performance and improved interaction in terms of speed between the client / server interaction.
In that manner of thought I recently had to enable the mod_deflate on CentOS release 5.4 with Apache webserver version httpd-2.2.3-31.

Here is an uname output to be more specific about the kernel release on which the Webserver is running:

Linux centos 2.6.18-128.7.1.el5 #1 SMP Mon Aug 24 08:21:56 EDT 2009 x86_64 x86_64 x86_64 GNU/Linux

CentOSâ€™s Apache webserver comes with enabled by default **deflate Apache module**.
Thus all necessery to be done in order to enable the http compression is to simply create the file and paste in it:

```
# Insert filter
SetOutputFilter DEFLATE

# Netscape 4.x has some problems...
BrowserMatch ^Mozilla/4 gzip-only-text/html

# Netscape 4.06-4.08 have some more problems
BrowserMatch ^Mozilla/4.0[678] no-gzip

# MSIE masquerades as Netscape, but it is fine
BrowserMatch bMSIE !no-gzip !gzip-only-text/html

# Don't compress images
SetEnvIfNoCase Request_URI .(?:gif|jpe?g|png)$ no-gzip dont-vary

# Don't compress already compressed stuff !
SetEnvIfNoCase Request_URI .(?:exe|t?gz|zip|bz2|sit|rar)$ no-gzip dont-vary
SetEnvIfNoCase Request_URI .pdf$ no-gzip dont-vary

# Make sure proxies don't deliver the wrong content
Header append Vary User-Agent env=!dont-vary
```

```
# Log Stuff !
# DeflateFilterNote Input input_info
# DeflateFilterNote Output output_info
# DeflateFilterNote Ratio ratio_info
# LogFormat '"%r" %{output_info}n/%{input_info}n (%{ratio_info}n%%)' deflate
# CustomLog /var/log/httpd/deflate_log deflate
```

I used the article HTTP Compression on Redhat / CentOS / Fedora to create this one, so thanks to the author of the up mentioned article.