

当前主题: [互联网络](#)

deflate——过时的网页压缩格式，最好禁用

在设置GZip时，发现同时有个Deflate压缩设置，一开始并不了解Deflate压缩，于是便在启用GZip的同时，也启用了Deflate压缩。虽然同时设置GZip和Deflate压缩，并不影响网站的正常运行，并且在检测网站是否启用GZip时，返回的答案是已启用。但是我们能否不设置Deflate压缩呢？或者说，需要禁用Deflate压缩呢？Deflate压缩又有什么好处和坏处？带着这一堆疑问，我今天查了一些资料，大概了解了一些，最后得到的结论是：DEFLATE——过时的网页压缩格式，最好禁用。

了解GZip和deflate

GZIP，好像是一个不透明的或原子的功能。事实上，HTTP定义了一种机制，一个Web客户机和Web服务器同意一压缩方案可以用来发送内容，这由使用Accept-Encoding和Content-Encoding标头完成。有两种常用的HTTP压缩：DEFLATE和GZIP。

DEFLATE是一个无专利的压缩算法，它可以实现无损数据压缩，有众多开源的实现算法。该标准的实施库大多数人用的是zlib的。zlib库提供用于压缩和解压缩使用DEFLATE/INFLATE的数据。zlib库还提供了一种数据格式，混淆的命名ZLIB，它包装DEFLATE压缩数据，具有报头和校验和。

GZIP是使用DEFLATE进行压缩数据的另一个压缩库。事实上，GZIP的大多数实现实际使用zlib库的内部进行DEFLATE/INFLATE压缩操作。GZIP产生其自己的数据格式，混淆的命名GZIP，它包装DEFLATE压缩数据，具有报头和校验和。

早期浏览器对DEFLATE压缩描述混乱

HTTP/1.1 RFC（超文本传输协议HTTP/1.1版）在为Accept-Encoding和Content-Encoding标头描述允许的压缩方案时做得不好，它定义了Content-Encoding: gzip，该响应体由使用GZIP数据格式（GZIP标头，压缩数据，和校验和）组成。它还定义Content-Encoding: DEFLATE，但是，尽管它的名字，这并不意味着响应体是DEFLATE压缩数据的原始块。根据RFC-2616，DEFLATE和Content-Encoding: DEFLATE，实际上意味着响应体是由zlib的格式（zlib的头，压缩数据和校验）组成的。

这种“DEFLATE的标识并不意味着原始DEFLATE压缩数据”的想法是相当混乱的。早期版本的Microsoft的IIS Web服务器被编程为返回原始DEFLATE压缩数据的Accept-Encoding: deflate要求，而不是一个zlib的格式响应。而在与Content-Encoding时期望的响应自然版本的Internet Explorer: DEFLATE标头有原始DEFLATE响应主体。

Mark Adler，zlib的作者之一，介绍说：

早期的微软服务器会错误地提供“Deflate”的原始压缩（例如RFC1951的数据没有RFC1950的zlib包装）。这导致的问题是，浏览器不得不去试一下两种方式，在最后它只是使用更可靠的GZIP。

现在浏览器对DEFLATE压缩处理不好

如Mark所说，浏览器收到Content-Encoding后，压缩必须处理两种可能的情况：响应主体是原始DEFLATE数据，或响应主体是zlib包装过的DEFLATE响应。那么，现代浏览器处理DEFLATE或zlib

包装过的DEFLATE响应效果有多好？Verve工作室测试了一个庞大的浏览器数量，结果并不好。

表中的部分结果意味着浏览器处理原始DEFLATE或zlib包装过的DEFLATE不一致，这其实是另一种说法“它被破坏了，功能不正常了。”这似乎是一个棘手的错误，而浏览器创造者不断重新引入到他们的产品中。Safari浏览器5.0.2？没问题。Safari浏览器5.0.3？彻底失败。Safari浏览器5.0.4？没问题。Safari浏览器5.0.5？不一致和被破坏。

发送原始DEFLATE数据不是一个好主意，正如马克说，“[它]只使用更可靠的GZIP。”

还应该注意的，所有支持DEFLATE的浏览器都支持GZIP，但不是所有支持GZip的浏览器都支持DEFLATE。有些浏览器，如Android，在它们的Accept-Encoding请求头不包含deflate压缩。由于你不得不配置你的Web服务器使用GZIP，你还不如避免Content-Encoding: deflate。

禁用DEFLATE

幸运的是，避免DEFLATE并不那么困难。

Apache处理所有HTTP压缩的模块是mod_deflate模块。尽管它的名字，mod_deflate模块根本不支持deflate。要得到一个发送原始DEFLATE或zlib包装过的DEFLATE的Apache2版本是不可能。Nginx的，如Apache，不支持deflate的，它只会发送gzip压缩的响应，发送Accept-Encoding: deflate请求头将导致未压缩的响应。

微软的IIS Web服务器可以同时发送gzip和deflate的响应，你可以分别启用或禁用每个方案。对于IIS6，你可以编辑metabase来禁用DEFLATE支持。对于IIS7，您可以通过编辑.config文件里的<httpCompression>元素的<schemes>的DEFLATE压缩方案部分禁用DEFLATE。

如果你的Web服务器发送压缩的DEFLATE内容，无论免费还是商业的产品都具有内置的检测，“过时的压缩格式”。

检测网页使用GZip还是deflate压缩

检测网页使用使用GZip还是deflate压缩，可以借助一些在线工具，例如卡卡网的[GZip压缩检测](http://pagespeed.webkaka.com/youhua/gzip/) (http://pagespeed.webkaka.com/youhua/gzip/) 。

gzip压缩检测

网址: 提交

浏览器: MSIE: Mozilla/4.0 (compatible; Windows ...

网址 <http://www.webkaka.com/> 的检测结果:

是否压缩	是
压缩类型	gzip
原文件大小	76221 字节
压缩后文件大小	15486 字节
压缩率	79.68%

Header信息:

Cache-Control	max-age=86400
Content-Length	15486
Content-Type	text/html
Content-Encoding	gzip
Content-Location	http://www.webkaka.com/index.html
Last-Modified	Sun, 08 Feb 2015 00:29:11 GMT
Accept-Ranges	bytes
ETag	"806519423643d01:148f1"
Vary	Accept-Encoding
Server	Microsoft-IIS/6.0
X-Powered-By	ASP.NET
Date	Mon, 09 Feb 2015 22:00:41 GMT

GZip压缩检测

需要特别注意的是，GZip压缩是按文件分别压缩的，并不是检测到网页启用了GZip，所有文件就全部启用GZip了。很多情况下，网页启用了GZip，但发现JS文件并没有启用GZip，这是因为设置不当而造成的。

如何知道网页的哪些文件启用了GZip，哪些文件还没有启用GZip？可以借助卡卡网的[网站速度诊断工具](http://pagespeed.webkaka.com/) (<http://pagespeed.webkaka.com/>)，一键查看。例如，看看下面的检测结果：

时间线 对象分析 网页统计 性能优化

• 使用浏览器缓存

• 启用GZIP压缩html

• 启用GZIP压缩css

• 启用GZIP压缩JavaScript

• 将图片合并为CSS贴图

• 优化CSS代码

• 优化JavaScript代码

• 优化图片

• 指定"Vary: Accept-Encoding"标头

启用GZIP压缩Html文件

启用gzip或deflate压缩资源可以减少网络传输的字节数

更多>

优化建议

用gzip工具压缩html文件可以减少 69.7kB (77.8%)的数据传输量

- <http://www.webkaka.com/index.html>

原体积: 75974Byte 压缩后: 14917Byte 减少: 61057Byte (80.4%)

✓已经启用Gzip, 压缩类型: gzip, Gzip压缩检测

html网页已经成功启用了GZip压缩



css文件未成功启用GZip压缩

这是因为，启用GZip压缩时只对html网页有效，而对其他文件如css没有生效。

您可能对如下文章也感兴趣

[图片使用GZip压缩后体积变大，慎用！](#)

[Nginx启用Gzip压缩js无效的原因](#)

[IIS启用GZIP压缩css、js无效的原因及解决方法](#)

[IIS启用GZip失败之原因：临时目录权限没设好](#)

[gzip压缩启动后js css不能运行的解决方法](#)

[网站启用GZip压缩后，速度快了3倍！](#)

标签: [gzip](#)

- 扩展阅读
- [⊙Nginx启用Gzip压缩js无效的原因](#) (2014-9-12 17:40:16)
- [⊙bmp图片使用GZip压缩率竟高达98.83%](#) (2014-6-10 16:39:55)
- [⊙gzip压缩启动后js css不能运行的解决方法](#) (2013-9-2 21:53:36)
- [⊙图片GZip压缩后体积竟然变大了](#) (2013-7-19 11:44:6)
- [⊙IIS启用GZip失败之原因：临时目录权限没设好](#) (2013-7-7 15:25:59)
- [⊙IIS启用GZIP压缩css、js无效的原因及解决方法](#) (2013-7-6 14:7:23)
- [⊙IIS启用GZip压缩的详细教程【图解】](#) (2013-7-4 17:21:11)