链滴

Nginx 静态资源 WEB 服务的应用一

作者：Weidong

原文链接：https://ld246.com/article/1520255235750

来源网站：链滴

什么是静态资源

所谓静态资源就是非服务器动态运行生产的文件，常见的静态资源类型有一下几种：

- 浏览器端渲染　HTML, CSS, JS
- 图片　JPEG, GIF, PNG 等
- 视频　FLV, MPEG 等
- 文件　TXT, 等下载文件

什么是动态资源

客户端发起一个动态资源请求时需要通过服务端解释器进行运算封装后再返回给客户端。

高效的文件读取机制 sendfile

sendfile 是 Linux 系统中的一种读取文件的系统调用（system call）机制，他的目的是为了减少数据内核空间与用户空间之间的复制次数和上下文切换次数，从而达到高效的文件读取效率。

在 Nginx 中就使用到了 sendfile 机制，这也是 Nginx 作为静态资源服务器性能更好的原因之一。

在 Nginx 中 sendfile 指令被包含在 http_core_module 模块中，所以他是一个核心指令。

sendfile 指令配置语法

```
Syntax:    sendfile on | off;
Default:    sendfile off;
Context:    http, server, location, if in location
```

用于开启或关闭 sendfile() 机制，默认为关闭。

他可以在 http, server, location, if in location 上下文中配置。

tcp_nopush 指令配置语法

```
Syntax:    tcp_nopush on | off;
Default:    tcp_nopush off;
Context:    http, server, location
```

用于开启或关闭 tcp_nopush，默认为关闭；

他可以在 http, server, location 上下文中配置。

tcp_nopush 是用于提高网络包的传输效率，他需要在开启 sendfile 的情况下才有用。那他是如何提传输效率的呢？

其实他是使用了 Linux socket 选项中的 TCP_CORK 选项。CORK 可以翻译成塞子的意思，当有数据时就将他塞住，不让他出去，等到几个数据包凑一块儿了才将塞子拔了让数据发送出去，以达到高效用网络资源的目的。

tcp_nodelay 指令配置语法

Syntax: tcp_nodelay on | off;
Default: tcp_nodelay on;
Context: http, server, location

用于开启或关闭 tcp_nodelay，默认为开启；

他可以在 http, server, location 上下文中配置。

当有数据需要响应包需要发送出去时就立即发送，提高数据包的传输实时性，对于实时性要求较高的景需要开启，他需要在开启 keepalive 模式下才会有效。

http_gzip_module

http_gzip_module 模块是一个使用 gzip 压缩方法对响应内容进行过滤压缩后响应给客户端以达到低网络消耗的模块。

由于响应给客户端的是一个压缩包，所有这需要浏览器支持解压，好在目前市面上的浏览器基本都支解压。

gzip 指令配置语法

Syntax: gzip on | off;
Default: gzip off;
Context: http, server, location, if in location

用于开启或关闭 gzip 压缩，默认为关闭状态；

他可以在 http, server, location, if in location 上下文中配置

gzip_buffers 指令配置语法

Syntax: gzip_buffers number size;
Default: gzip_buffers 32 4k|16 8k;
Context: http, server, location

设置用于解压缩响应的缓冲区的 number 和 size 。默认情况下，缓冲区大小等于一个内存页面，4K 8K，取决于操作系统平台

gzip_comp_level 指令配置语法

Syntax: gzip_comp_level level;
Default: gzip_comp_level 1;
Context: http, server, location

用于设置压缩级别，可压缩级别为 1-9；

他可以在 http, server, location 上下文中配置。

需要主要的是压缩比越高他所占用的 CPU 消耗就越高，所以应该根据不同的文件类型设置不同的压比。

gzip_http_version 指令配置语法

Syntax: gzip_http_version 1.0 | 1.1;
Default: gzip_http_version 1.1;
Context: http, server, location

用于控制压缩功能所要求的最低 HTTP 协议版本，默认为 1.1；
他可以在 http, server, location 上下文中配置。

gzip_disable 指令配置语法

Syntax: gzip_disable regex ...;
Default: —
Context: http, server, location

用于设置 gzip_disable 禁用条件；通过 REGEX （正则表达式）对 “User-Agent” 进行匹配。
他可以在 http, server, location 上下文中配置。

gzip_min_lenght 指令配置语法

Syntax: gzip_min_length length;
Default: gzip_min_length 20;
Context: http, server, location

设置可以被压缩的响应字段的最小长度；
他可以在 http, server, location 上下文中配置。

gzip_proxied 指令配置语法

Syntax: gzip_proxied off | expired | no-cache | no-store | private | no_last_modified | no_etag auth | any ...;
Default: gzip_proxied off;
Context: http, server, location

用于根据请求和响应，启用或禁用代理请求和响应，他通过请求头部中的 Via 字段的存在来确定；

off 表示禁用所有代理请求的压缩；
expired 表示如果响应头中包含 Expired 字段时启用压缩；
no-cache 表示如果响应头中包含 Cache-Control 字段 no-cache 参数时启用压缩；
no-store 表示如果响应头中包含 Cache-Control 字段 no-store 参数时启用压缩；
private 表示如果响应头中包含 Cache-Control 字段 private 参数时启用压缩；
no_last_modified 表示如果响应头中不包含 Last_Modified 字段时启用压缩；
no_etag 表示如果响应头中不包含 ETag 字段时启用压缩；
auth 表示如果请求头中包含 auth 字段时启用压缩；
any 表示启用所有代理请求的压缩。
他可以在 http, server, location 上下文中配置。

gzip_types 指令配置语法

Syntax: gzip_types mime-type ...;
Default: gzip_types text/html;
Context: http, server, location

对指定的 MIME 类型进行压缩，如果使用 * 则表示对任何 MIME 类型都压缩；对于 text/html 类型终会压缩。

他可以在 http, server, location 上下文中配置。

MIME (Multipurpose Internet Mail Extensions) 多用途互联网邮件扩展类型,是秒速消息内容类型互联网标准。

http_gzip_static_module

http_gzip_static_module 是一个用于实现 gzip 预读的模块；也就是 Nginx 进程在读取磁盘上的某文件时会先检查是否有这个文件的 .gz 压缩版本存在，如果有就将压缩版本响应给客户端。

这个模块默认是不会被编译的，如需要编译使用 --with-http_gzip_static_module

gzip_static 指令配置语法

Syntax: gzip_static on | off | always;
Default: gzip_static off;
Context: http, server, location

用于开启过关闭 gzip 预读功能，默认为关闭；

他可以在 http, server, location 上下文中配置。

http_gunzip_module

http_gunzip_module 是用于解决客户端不支持 gzip 压缩编码时使用 gunzip 方法进行压缩响应。

默认不被编译，编译参数 --with-http_gunzip_module

gunzip 指令配置语法

Syntax: gunzip on | off;
Default: gunzip off;
Context: http, server, location

用于开启或关闭 gunzip 功能，默认为关闭；

他可以在 http, server, location 上下文中配置。

gunzip_buffers 指令配置语法

Syntax: gunzip_buffers number size;
Default: gunzip_buffers 32 4k|16 8k;
Context: http, server, location

设置用于解压缩响应的缓冲区的 number 和 size 。默认情况下，缓冲区大小等于一个内存页面，4K

8K，这取决于操作系统平台。

他可以在 http, server, location 上下文中配置。