1、可针对静态资源高速高并发访问及缓存 2、可使用反向代理加速,并且可进行数据缓存 3、具有简单负载均衡、节点健康检查和容错功能 4、支持远程FastCGI服务的缓存加速 nginx可以做什么 5、支持FastCGI、Uwsgi、SCGI、Memcached Servers的加速和缓存 6、支持SSL、TLS、SNI 7、具有模块化的架构: 过滤器包括gzip压缩、ranges支持、chunked响应 XSLT、SSI及图像缩放等功能。在SSI过滤器中,一个包含多个SSI的页面 如果经由FastCGI或反向代理处理,可被并行处理 1、支持基于名字、端口及IP的多虚拟主机站点 2、支持Keep-alive和pipelined连接 3、可进行简单、方便、灵活的配置和管理 4、支持修改Nginx配置,并且在代码上线时,可平滑重启,不中断业务访问 5、可自定义访问日志格式,临时缓冲写日志操作,快速日志轮询及通过rsyslog处理日志 6、可利用信号控制Nginx进程 7、支持3xx-5xx HTTP状态码重定向 nginx作为WEB服务器的支持 8、支持rewrite模块,支持URI重写及正则表达式匹配 9、支持PUT、DELETE、MKCOL、COPY及MOVE等较特殊的HTTP请求方法 10、支持基于客户端IP地址和HTTP基本认证的访问控制 11、支持FLV流和MP4流技术产品应用 12、支持HTTP响应速率限制 13、支持同一IP地址的并发连接或请求数限制 14、支持邮件服务代理 1、作为web服务软件 2、反向代理或负载均衡服务 location ~ .*\.(gif|jpg|jpeg|png|bmp|swf)\$ expires nginx的应用场景 location ~ .*\.(js|css)?\$ 12h; expires 3、前端业务数据缓存服务 1、支持高并发:能支持几万并发连接(特别是静态小文件业务环境) 2、资源消耗少:在3万并发连接下,开启10个Nginx线程消耗的内存不到200MB 3、可以做HTTP反向代理及加速缓存,即负载均衡功能,内置对RS节点服务器 健康检查功能,这相当于专业的Haproxy软件或LVS的功能 nginx的优点 4、具备Squid等专业缓存软件等的缓存功能 5、支持异步网络I/O事件模型epoll (Linux 2.6+) Nginx服务 nginx的核心功能模块 (Core functionality) 1、Apache2.2版本非常稳定强大,据官方说,Apache2.4版本性能更强 2、Prefork模式取消了进程创建开销,性能很高 3、处理动态业务数据时,因关联到后端的引擎和数据库,瓶颈不在Apache上 4、高并发时消耗系统资源相对多一些 5、基于传统的select模型,高并发能力有限。select 模型 也就是 同步epoll 异步 6、支持扩展库,可通过DSO、apxs方法编译安装额外的插件功能,不需要重新编译Apache apache服务的特点 7、功能多, 更稳定, 更安全, 插件也多 BUT: 8、市场份额在逐年递减 每个请求的状态始终在维护着 (消耗好多资源) 同步网络模式 select 网络模式: epoll 处理服务器端的并发 请求人数越多服务器肯定吃紧 系统资源也会紧张 I/O效率也会很慢 异步网络模式 1、基于异步网络I/O模型,性能、并发都与Nginx相近 2、扩展库是SO模式,比Nginx灵活 3、目前国内的使用率比较低,安全性没有Apache和Nginx好 lighttpd服务的特点 4、通过插件 (mod secdownload) 可实现文件URL地址加密 (优点) BUT: 5、社区不活跃,市场份额较低 若是高并发场景,尽量采用Nginx或Lighttpd,二者首选Nginx ___理论上采用Nginx和Δpache均可,建议选择Nginx,为了避免相同业务的服务教件多样化增加额外维护成本。动态业务可以由Nginx兼做前端代理,再根据页面元素的类型或目录转发到后端相应的服务器进行处理 采用Nginx 如何选择web服务器 既有静态业务又有动态业务 如果并发不是很大,又对Apache很熟悉,采用Apache也是可以的,Apache 2.4版本也很强大,并发连接数也有所增加 服务器选择总结: 在工作中不要盲目选择软件,这可能最终会导致自己无法控制局面,从而给企业带来灾难性的损失 nginx安装和虚拟站点配置方式 详情见: Nginx.md 负载均衡服务的配置 详情见: Nginx.md 是nginx支持的负载均衡的模块 upstream模块 详情见: Nginx.md