# 1 Nginx

# 2 Tengine

http://tengine.taobao.org/

Tengine是由淘宝网发起的Web服务器项目。它在Nginx的基础上，针对大访问量网站的需求，添加了很多高级功能和特性。Tengine的性能和稳定性已经在大型的网站如淘宝网，天猫商城等得到了很好的检验。它的最终目标是打造一个高效、稳定、安全、易用的Web平台。

从2011年12月开始，Tengine成为一个开源项目，Tengine团队在积极地开发和维护着它。Tengine团队的核心成员来自于淘宝、搜狗等互联网企业。Tengine是社区合作的成果，我们欢迎大家参与其中，贡献自己的力量。

## 2.1 ngx\_http\_upstream\_dynamic\_module

此模块提供了在运行时动态解析upstream中server域名的功能。

upstream backend {

dynamic\_resolve fallback=stale fail\_timeout=30s;

server a.com;

server b.com;

}

server {

...

location / {

proxy\_pass http://backend;

}

}

指令

Syntax: dynamic\_resolve [fallback=stale|next|shutdown] [fail\_timeout=time]

Default: -

Context: upstream.

指定在某个upstream中启用动态域名解析功能。

fallback参数指定了当域名无法解析时采取的动作：

fail\_timeout参数指定了将DNS服务当做无法使用的时间，也就是当某次DNS请求失败后，假定后续多长的时间内DNS服务依然不可用，以减少对无效DNS的查询。

# 3 OpenResty

<http://openresty.org/cn/> ,更新至2018年9月18日

OpenResty® 是一个基于 Nginx 与 Lua 的高性能 Web 平台，其内部集成了大量精良的 Lua 库、第三方模块以及大多数的依赖项。用于方便地搭建能够处理超高并发、扩展性极高的动态 Web 应用、Web 服务和动态网关。

OpenResty® 通过汇聚各种设计精良的 Nginx 模块（主要由 OpenResty 团队自主开发），从而将 Nginx 有效地变成一个强大的通用 Web 应用平台。这样，Web 开发人员和系统工程师可以使用 Lua 脚本语言调动 Nginx 支持的各种 C 以及 Lua 模块，快速构造出足以胜任 10K 乃至 1000K 以上单机并发连接的高性能 Web 应用系统。

OpenResty® 的目标是让你的Web服务直接跑在 Nginx 服务内部，充分利用 Nginx 的非阻塞 I/O 模型，不仅仅对 HTTP 客户端请求,甚至于对远程后端诸如 MySQL、PostgreSQL、Memcached 以及 Redis 等都进行一致的高性能响应。

京东是我们在国内最大的一个 OpenResty 用户，你在京东上看到的所有商品详情页的调用都是基于 OpenResty 做的。还有新浪财经、新浪股票和新浪体育所有的后端，包括360搜索页面的跳转，同样都是基于 OpenResty。在国外大家比较常用的 Github、CloudFlare 也都有用 OpenResty 做一些服务。

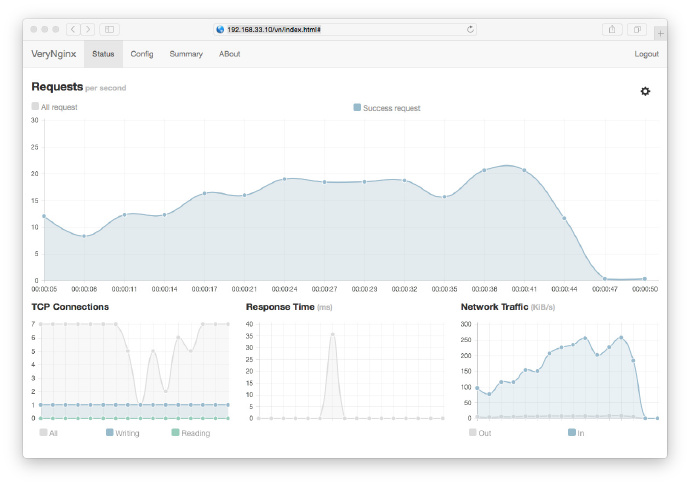
# 4 VeryNginx

<https://gitee.com/alexa/VeryNginx> , 更新至2018年9月29日

VeryNginx 基于 Lua\_Nginx\_Module(openrestry) 开发，实现了高级的防火墙、访问统计和其他的一些功能。 强化了 Nginx 本身的功能，并提供了友好的 Web 交互界面。

Nginx 运行状态分析

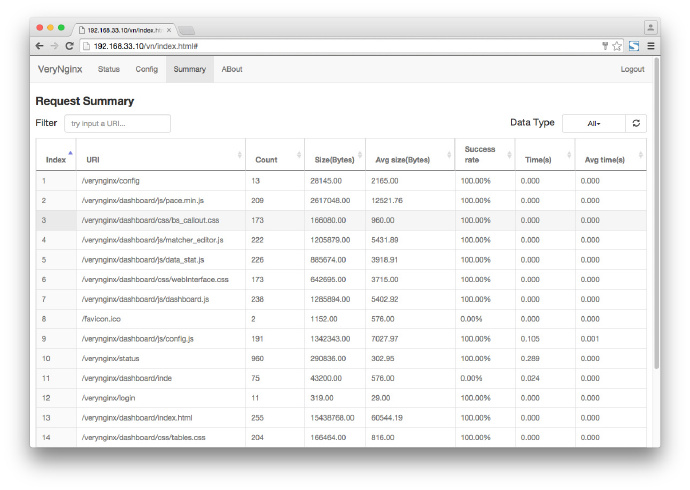
* 每秒请求数
* 响应时间
* 网络流量
* 网络连接数



访问统计

VeryNginx 可以统计网站每个URI的访问情况，包括每个URI的:

* 总请求次数
* 各状态码次数
* 返回总字节数
* 每请求平均字节数
* 总响应时间
* 平均响应时间



# 5 Kong

<https://konghq.com/kong-community-edition/>

API 网关，即API Gateway，是大型分布式系统中，为了保护内部服务而设计的一道屏障，可以提供高性能、高可用的 API托管服务，从而帮助服务的开发者便捷地对外提供服务，而不用考虑安全控制、流量控制、审计日志等问题，统一在网关层将安全认证，流量控制，审计日志，黑白名单等实现。网关的下一层，是内部服务，内部服务只需开发和关注具体业务相关的实现。网关可以提供API发布、管理、维护等主要功能。开发者只需要简单的配置操作即可把自己开发的服务发布出去，同时置于网关的保护之下。

Kong是一款基于Nginx\_Lua模块写的高可用，易扩展由Mashape公司开源的API Gateway项目。由于Kong是基于Nginx的，所以可以水平扩展多个Kong服务器，通过前置的负载均衡配置把请求均匀地分发到各个Server，来应对大批量的网络请求。

Kong主要有三个组件：

* Kong Server ：基于nginx的服务器，用来接收API请求。
* Apache Cassandra/PostgreSQL ：用来存储操作数据。
* Kong dashboard：官方推荐UI管理工具，当然，也可以使用 restfull 方式 管理admin api。

Kong采用插件机制进行功能定制，插件集（可以是0或n个）在API请求响应循环的生命周期中被执行。插件使用Lua编写，目前已有[几个基础功能](http://getkong.org/plugins/" \t "_blank)：HTTP基本认证、密钥认证、CORS（ Cross-origin Resource Sharing，跨域资源共享）、TCP、UDP、文件日志、API请求限流、请求转发以及nginx监控。

Kong 官网：[https://getkong.org/](https://getkong.org/" \t "_blank) 插件介绍，帮助文档等。

Kong 源码：[https://github.com/Mashape/kong](https://github.com/Mashape/kong" \t "_blank)

Kong UI管理工具：[https://github.com/PGBI/kong-dashboard](https://github.com/PGBI/kong-dashboard" \t "_blank) 浏览器UI管理工具

Kong 桌面管理工具：[https://github.com/ajaysreedhar/kongdash](https://github.com/ajaysreedhar/kongdash" \t "_blank) 有linux版本，windows版本，mac版本

# 6 Orange

<http://orange.sumory.com/docs/> ,更新至2017年5月16日

Orange是一个基于OpenResty的API Gateway，提供API及自定义[规则](http://orange.sumory.com/docs/rule.html)的监控和管理，如访问统计、流量切分、API重定向、API鉴权、WEB防火墙等功能。Orange可用来替代前置机中广泛使用的Nginx/OpenResty， 在应用服务上无痛前置一个功能丰富的网关系统。它有以下特性：

* 动态更新Nginx/OpenResty配置而无需重启或reload
* 通过MySQL存储来简单支持集群部署
* 支持多种[条件匹配和变量提取](http://orange.sumory.com/docs/expression.html)
* 支持通过自定义插件方式扩展功能
* 内置多个通用插件
  + 全局状态统计
  + 自定义监控
  + URL重写
  + URI重定向
  + 访问限速
  + Key based rate limiting
  + HTTP Basic Auth
  + HTTP Key Auth
  + Signature Auth
  + 简单防火墙WAF
  + 代理、ABTesting、分流
  + Shared Dict存取接口
* 提供Dashboard用于管理内置插件
* 开放API: 所有插件均开放API供第三方使用， 通过这些API可简单灵活得配置插件、查看运行状态、统计数据等。
* 架构简单，依赖少，许可协议宽松，适合直接二次开发或改造。

# 7 Vanilla

<https://github.com/idevz/vanilla>

Vanilla 是新浪移动事业部系统架构组的同学基于 OpenResty 自主研发的一个 Web 开发框架，目前服务于新浪移动后端多条产品线及核心接口池。

更新至2017年11月2日