



# 从头搭建Nginx静态资源服务

# Nginx是什么？

- **Web场景的高性能**解决方案
  - 静态资源服务：高效的磁盘IO处理能力
    - CDN
  - 七层/四层负载均衡：高效的网络协议处理能力
    - 正向代理
    - 反向代理
  - 整合Lua语言的负载均衡：丰富而完善的生态
    - API网关
    - Waf防火墙

# Nginx有什么特点？

- 高性能
  - 高并发 (C10M)
  - 低时延
- 稳定
  - mainline/stable版本
  - master/worker进程结构
- 开放的生态
  - BSD License: [Tengine](#)
  - 拥有极高自由度的C模块: [Openresty](#)、[Kong](#)

# 应当如何获取、安装Nginx?

- 放弃定制化能力
  - docker镜像
  - apt-get/yum安装
  - 直接下载编译好的二进制文件
- 定制化Nginx/Openresty
  - [nginx.org/nginx-cn.net](https://nginx.org/nginx-cn.net)下载源代码
    - 下载第三方模块源代码
  - configure/make/make install

# 如何确定该选择哪个Nginx版本？

- CHANGES
  - bugfix
  - feature
  - change
  - security
- 你的Nginx用的是哪个版本？
  - `nginx -v`

# 为什么需要定制化?

- 默认configure定制后的Nginx, 没有哪些功能?
  - 不支持http2
  - 不支持TCP/UDP协议负载均衡
  - 不支持stub\_status性能监控
  - 不支持TLS/SSL安全协议
  - ... ..
- Nginx为什么要设计编译时的定制化功能?
  - 提升性能
  - 减小可执行文件的体积
  - 有些功能, 依赖许多软件, 准备环境较复杂
  - 强大的自定义功能, 包括编译、运行时各种路径、参数的指定

# 怎样编译、安装、启动Nginx?

1. 认识Nginx源代码目录
2. configure定制Nginx的编译、运行环境
3. make编译nginx
4. make install安装
5. 启动nginx

# configure的用法

- 你的Nginx用了哪些configure选项?
  - nginx -V
- configure步骤
  - 解析configure参数, 生成编译参数: auto/options
    - 第三方模块通过--add-module参数会添加模块至NGX\_ADDONS变量
  - 针对不同操作系统、体系架构、编译器, 选择特性 (例如linux中的epoll或者windows中的iocp) 及生成相应编译参数
  - 根据所有模块生成ngx\_modules.c及makefile
  - 在屏幕上显示configure执行结果: auto/summary



# nginx.conf语法格式

- 指令以;符号结尾，以空格分离参数
- 指令块包裹在{}中
- #是注释
- \$是变量
- 各指令以不同方式支持正则表达式
- include可以读入新文件，方便维护

# 配置参数的单位

- 时间
  - ms: 毫秒
  - s: 秒
  - m: 分钟
  - h: 小时
  - d: 天
  - w: 周
  - M: 月
  - y: 年
- 空间
  - k/K: KB
  - m/M: MB
  - g/G: GB

```
http {
    include      mime.types;
    upstream thwp {
        server 127.0.0.1:8000;
    }

    server {
        listen 443 http2;
        #Nginx配置语法
        limit_req_zone $binary_remote_addr zone=one:10m rate=1r/s;
        location ~* \.(gif|jpg|jpeg)$ {
            proxy_cache my_cache;
            expires 3m; proxy_cache_key $host$uri$is_args$args;
            proxy_cache_valid 200 304 302 1d;
            proxy_pass http://thwp;
        }
    }
}
```

# nginx.conf在哪里？

- `nginx -c`: 命令行指定配置文件路径
- `configure --prefix`: 编译时指定
- `configure --conf-path`: 编译时指定

# 如何让vim “有颜色” 的显示nginx.conf?

- 复制contrib/vim/\*至 ~/.vim/

# 谢谢



# 活动推荐



api days Conferences Reports Initiatives API Landscape Media More

## INTERFACE

The first global digital API Conference  
June 30 & July 1, 2020

[REGISTER NOW](#) [APPLY TO BE A SPEAKER](#) [SPONSORSHIP DECK](#)

2 DAYS / 50 SPEAKERS / 5,000 ATTENDEES / 3,000 COMPANIES

<https://www.apidays.co/interface>



NGINX开源社区群



NGINX开源社区公众号



 **NGINX**  
Part of F5

### NGINX开源社区基础培训系列课程(第一季)

课程安排:  
每周四, 晚8:00-9:00

- ▶ 6月04日 从头搭建一个静态资源服务
- ▶ 6月11日 用好access.log日志
- ▶ 6月18日 NGINX的进程架构
- ▶ 6月25日 让NGINX提供HTTPS服务

  
讲师: 陶辉  
NGINX顶级专家



nginx-cn.net



NGINX开源社区公众号

NGINX开源社区正式登录中国，快来扫码关注，获得海量的NGINX干货吧。



NGINX开源社区群

扫描识别二维码进入群聊，加入大家庭！与大家交流讨论使用NGINX时的问题吧。



F5 社区公众号

关注公众号，您将获得关于F5的第一手资料，不错过任何一场干货讲座。

