

# 优极限

“极限教育，挑战极限”

[www.yjxxt.com](http://www.yjxxt.com)

极限教育，挑战极限。优极限是一个让 95% 的学生年薪过 18 万的岗前培训公司，让我们的学员具备优秀的互联网技术和职业素养，勇攀高薪，挑战极限。公司位于上海浦东，拥有两大校区，共万余平。累计培训学员超 3 万名。我们的训练营就业平均月薪 19000，最高年薪 50 万。

核心理念：让学员学会学习，拥有解决问题的能力，拿到高薪职场的钥匙。

项目驱动式团队协作、一对一服务、前瞻性思维、教练式培养模型-培养你成为就业明星。首创的老学员项目联盟给学员充分的项目、技术支撑，利用优极限平台这根杠杆，不断挑战极限，勇攀高薪，开挂人生。

扫码关注优极限微信公众号：

（获取最新技术相关资讯及更多源码笔记）



上一篇文章《FastDFS分布式文件系统详解》中带大家详细了解了它的核心概念，架构体系及 FastDFS 环境的搭建与使用。但是此时还无法通过 HTTP 对文件进行访问，这篇文章将带大家解决这个问题。

如果我们想通过 HTTP 访问 FastDFS 中保存的图片信息，需要借助其他工具实现，Nginx 就是一个不错的选择，它是一个高性能的 HTTP 和反向代理 Web 服务器。关于 Nginx 更多的内容请阅读《一篇文章搞定 Nginx 反向代理与负载均衡》。

# NGINX

那么问题来了：既然文件都上传至 Storage 服务器了，在 Storage 服务器中直接安装 Nginx 反向代理至文件资源路径，好像就能实现浏览器直接访问文件的效果。究竟是不是这样玩的呢？继续往下看。

其实真实环境中并不是这样使用的，因为 FastDFS 是**通过 Tracker 服务器将文件存储在 Storage 服务器中**，而且**同组存储服务器之间还需要进行文件复制**，会有**同步延迟**的问题。

假设 Tracker 服务器将文件上传到了 192.168.10.102，上传成功以后文件相关信息已经返回给客户端了。此时 FastDFS 的集群存储机制会将这个文件同步到同组中的其他机器上进行存储 192.168.10.103，在文件还没有复制完成的情况下，客户端如果用这个文件的相关信息在 192.168.10.103 上进行获取，就会出现文件无法访问的错误。

解决办法就是使用 `fastdfs-nginx-module`，它可以**重定向文件链接到源服务器**进行获取，避免客户端由于复制延迟导致的文件无法访问的错误。

## 配置 fastdfs-nginx-module

### 下载资源

Nginx 整合 FastDFS 时 Nginx 需要添加 `fastdfs-nginx-module` 模块。直接通过 Github: <https://github.com/happyfish100/fastdfs-nginx-module> 下载压缩包或者使用 git 命令下载，或者通过资源地址: <https://sourceforge.net/projects/fastdfs/files/> 下载。

下载以后，上传资源 `fastdfs-nginx-module-master.zip` 至服务器 `/usr/local/src` 目录后并解压。

```
unzip fastdfs-nginx-module-master.zip
```

## 修改配置

复制配置文件 `mod_fastdfs.conf` 到 `/etc/fdfs` 目录中。

```
cp /usr/local/src/fastdfs-nginx-module-master/src/mod_fastdfs.conf /etc/fdfs/
```

`vim /etc/fdfs/mod_fastdfs.conf` 编辑配置文件，主要关注以下部分。

```
# tracker 服务器的 IP 和端口
tracker_server = 192.168.10.101:22122
# url 地址是否包含组名/卷名
url_have_group_name = true
# 数据组/卷对应的路径地址
store_path0 = /fastdfs/storage/store
```

## 拷贝资源

复制 `fastdfs` 安装包中的两个配置文件 `http.conf` 和 `mime.types` 到 `/etc/fdfs` 目录中。

```
cp /usr/local/src/fastdfs-master/conf/http.conf /etc/fdfs/
cp /usr/local/src/fastdfs-master/conf/mime.types /etc/fdfs/
```

## 安装 Nginx

## 下载资源

下载 Nginx 并解压。

```
# 下载 nginx 压缩包
wget -P /usr/local/src http://nginx.org/download/nginx-1.18.0.tar.gz
# 解压
tar -zxvf /usr/local/src/nginx-1.18.0.tar.gz -C /usr/local/src
```

## 安装依赖

Nginx 是基于 C 语言开发的，HTTP 模块使用了 `pcre` 来解析正则表达式，且可以使用 `zlib` 对 HTTP 包的内容进行 `gzip` 压缩，Nginx 不仅支持 HTTP 协议，还支持 HTTPS，HTTPS 需要 `openssl` 提供支持，所以安装 Nginx 之前必须先安装它所依赖的环境。

```
yum install -y gcc gcc-c++ pcre pcre-devel zlib zlib-devel openssl-devel
```

## 安装 Nginx

编译并安装。

```
# 切换至 nginx 的解压目录
cd /usr/local/src/nginx-1.18.0
# 创建 nginx 的安装目录
mkdir -p /usr/local/nginx
# 添加 fastdfs-nginx-module 模块, 指定 nginx 的安装路径
./configure --add-module=/usr/local/src/fastdfs-nginx-module-master/src --
prefix=/usr/local/nginx/
# 编译并安装
make && make install
```

## 配置 Nginx

`vim /usr/local/nginx/conf/nginx.conf` 编辑配置文件, 在 80 端口下添加以下内容。关于 Nginx 启动用户的问题请根据自身实际环境进行配置。

```
location ~/group[0-9]/ {
    ngx_fastdfs_module;
}
```

```
server {
    listen      80;
    server_name localhost;

    #charset koi8-r;

    #access_log  logs/host.access.log  main;

    location / {
        root     html;
        index    index.html index.htm;
    }

    location ~/group[0-9]/ {
        ngx_fastdfs_module;
    }

    #error_page  404              /404.html;
```

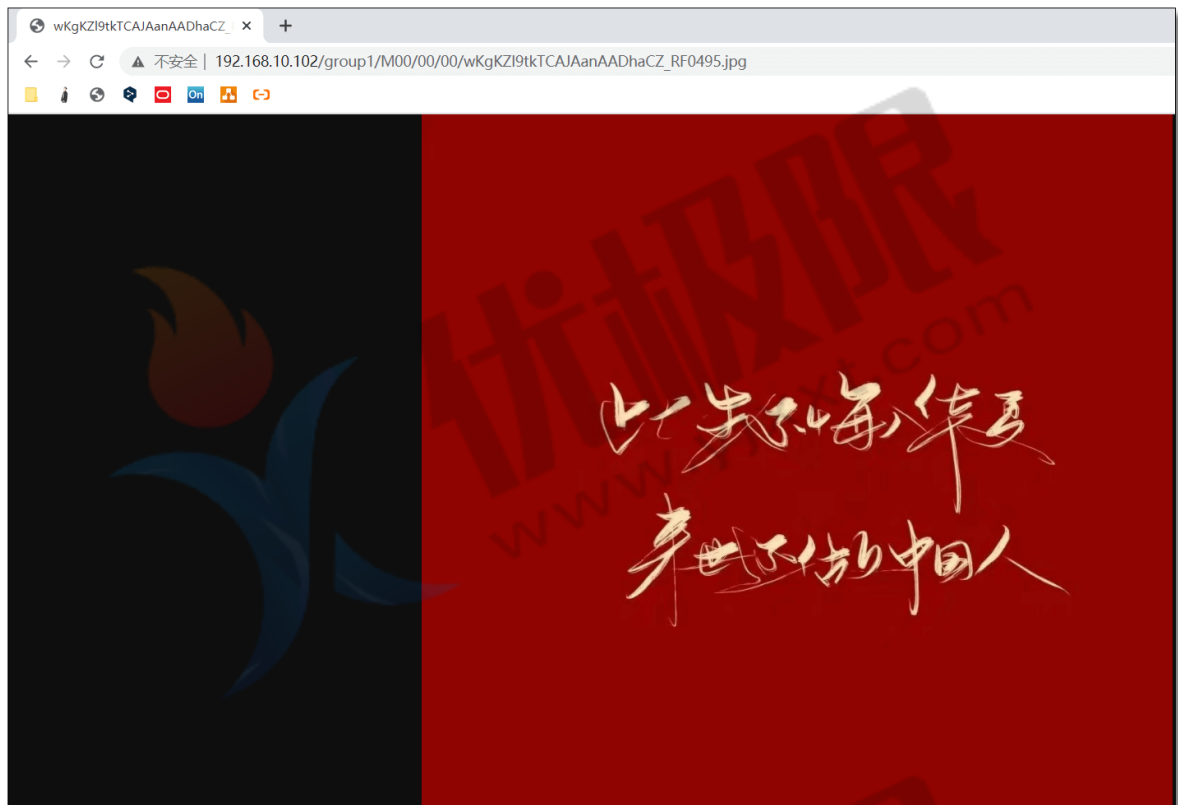
## 测试

启动 Nginx, 命令为: `/usr/local/nginx/sbin/nginx`

此时客户端上传图片以后得到文件名为:

`group1/M00/00/00/wKgKZ19tkTCAJAanAADhaCZ_RF0495.jpg`

浏览器访问: [http://192.168.10.102/group1/M00/00/00/wKgKZ19tkTCAJAanAADhaCZ\\_RF0495.jpg](http://192.168.10.102/group1/M00/00/00/wKgKZ19tkTCAJAanAADhaCZ_RF0495.jpg) 效果如下:



至此 Nginx 整合 FastDFS 实现文件服务器就到这里，其实目前的环境只是一个单节点的 FastDFS，如果实现高可用，Tracker 和 Storage 都需要搭建多个节点形成集群环境，下文就教大家搭建 FastDFS 集群环境。