

技术白皮书

技术白皮书.....	1
一、 nginx php windows 下配置.....	2
二 ubuntu 下面的 nginx php 安装.....	4
三 gitolite on ubuntu.....	5

— nginx php windows 下配置

1、从 nginx 网站下载 windows 版本的 nginx;windows 版本的 nginx 因为使用 socket select 方式，所以并发性能并不好，只能作为简单的开发环境使用。

2、从 php 网站下载 php for windows

3、修改 nginx.conf 文件，主要放开 php 的 fastcgi 配置部分：

```
location ~ \.php$ {  
    root          html;  
    fastcgi_pass   127.0.0.1:9000;  
    fastcgi_index  index.php;  
    fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script  
name;  
    include        fastcgi_params;  
}
```

特别是红线部分，是存放 php 文件的路径名称，配置不对，php-cgi 将找不到 php 文件。

4、php 目录下面的 php.ini.development 拷贝为 php.ini

5、创建 nginx 的批处理文件内容如下：

```
start c:\gexy\php-7.1.8-nts-Win32-VC14-x86\php-cgi.exe -b 127.0.0.1:9000  
start nginx.exe
```

先启动 php 在本地的 127.0.0.1:9000

然后启动 nginx

6、在 nginx 目录下面的 html/文件夹下创建 info.php 文件内容如下：

```
<?php phpinfo(); ?>
```

7、现在在浏览器中输入 127.0.0.1/info.php 将会显示 php 信息网页

8、其他一些 nginx 命令：

nginx -s t //测试 conf 文件是否正确

nginx -s reload //重新加载 nginx

nginx -s quit //退出

nginx -h //help

二 ubuntu 下面的 nginx php 安装

1、nginx 配置

```
ubuntu@VM-19-110-ubuntu: /etc/nginx/sites-available
2 | # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
3 | #
4 | location ~ \.php$ {
5 |     include snippets/fastcgi-php.conf;
6 |
7 |     # # With php7.0-cgi alone:
8 |     #     fastcgi_pass 127.0.0.1:9000;
9 |     #     fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
10 |    #     include fastcgi_params;
11 |    # # With php7.0-fpm:
12 |    fastcgi_pass unix:/run/php/php7.0-fpm.sock;
13 | }
14 | location ~ book.lp {
15 |     proxy_set_header Host www.brgguru.com;
16 |     proxy_pass http://127.0.0.1:8001;
17 | }
18 | location ~ \.lp$ {
19 |     proxy_set_header Host www.brgguru.com;
20 |     proxy_set_header X_Real_IP $remote_addr;
21 |     proxy_pass http://115.159.81.210;
22 | }
```

该配置为 php 配置，且把 lp 后缀的 lua 文件转发到 115.159.81.210 网站；同时把 book.lp 留在本机的 8001 网站处理，使用 skynet.

2、php 配置

```
ubuntu@VM-19-110-ubuntu: /etc/php/7.0/fpm/pool.d$ vi www.conf
```

重启php-fpm和nginx

```
root@tb:/etc/nginx/conf.d# service nginx reload
* Reloading nginx configuration nginx

root@tb:/etc/nginx/conf.d# service php7-fpm status
* php-fpm7.0 is running

root@tb:/etc/nginx/conf.d# service php7-fpm reload
* Reloading PHP 7.0 FastCGI Process Manager php-fpm7.0[ OK ]

root@tb:/etc/nginx/conf.d#
```

三 gitolite on ubuntu

1、安装步骤

gitolite on ubuntu

小刀客 5月11日 09:36

```
1 sudo adduser --system --shell /bin/sh --group --  
disabled-password --home /home/gitolite gitolite  
2 sudo apt-get -y install gitolite3  
3 sudo su - gitolite  
4 gitolite setup -pk yourname.pub
```

```
ubuntu@VM-19-110-ubuntu:/home/gitolite$ ls -l  
total 0  
-rw----- 1 gitolite gitolite  0 May 18 09:25 projects.list  
lrwxrwxrwx 1 gitolite gitolite 17 May 11 15:58 repositories -> /gxy/repositories
```

仓库选择软连接到其他文件夹下了。

2、管理用户和版本库

不应该手动在服务器端加入新的用户或者版本库。

gitolite 使用一个特殊的版本库 **gitolite-admin** 来管理用户和版本库，只要在这个版本库中修改并 **push**，服务器就会自动根据配置作出修改。

首先在客户端迁出版本库：

```
1 git clone git@host:gitolite-admin
```

如果在迁出的过程中询问密码，那么说明配置出了问题。一般情况是密钥配置错误。可以检查客户端的 `~/.ssh` 下有没有 **YourName** 私钥。如果需要使用不同的密钥连接多个 **ssh** 服务器，可以编辑 `~/.ssh/config` 进行配置。

进入 **gitolite-admin** 目录，其中的 **keydir** 目录是用来放置用户公钥的，

而 `conf/gitolite.conf` 则是用来配置用户和版本库。

编辑 `conf/gitolite.conf` 如下：

```
1 repo foo
```

```
2   RW+          =   alice
```

```
3   RW           =   bob
```

```
4   R            =   carol
```

上面的配置的含义是：

有一个名为 `foo` 的版本库；

用户 `alice` 对它有读、写、删除权限；

用户 `bob` 对它有读写权限；

用户 `carol` 对它仅有只读权限。

