搭建FireKylin个人博客

****1. 运行环境准备****

*安装 Node.js*

使用 yum 命令安装 Node.js

要通过 yum 来安装 nodejs 和 npm 需要先给 yum 添加 epel 源

添加 epel 源

64位:

rpm -ivh http://download.fedoraproject.org/pub/epel/6/x86\_64/epel-release-6-8.noarch.rpm

导入 key:

rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6

添加 remi 源

rpm -ivh http://rpms.famillecollet.com/enterprise/remi-release-6.rpm

rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-remi

安装完成后,执行

curl --silent --location https://rpm.nodesource.com/setup\_5.x | bash -

yum -y install nodejs

*使用 NPM 安装 PM2*

通过 NPM 安装进程管理模块 PM2。它是 Node.js 的一个进程管理模块，之后我们会使用它来管理我们的个人网站进程。

npm install pm2 -g

*安装 MySQL*

使用 yum 安装 MySQL：

wget http://dev.mysql.com/get/mysql-community-release-el7-5.noarch.rpm

rpm -ivh mysql-community-release-el7-5.noarch.rpm

yum install mysql-community-server -y

启动 MySQL 服务：

service mysqld restart

设置 MySQL 账户 root 密码：

/usr/bin/mysqladmin -u root password '4aE5aOii4Firekylin'

*安装 Nginx*

在 CentOS 上，可直接使用 yum 来安装 Nginx

1、添加源

　　默认情况Centos7中无Nginx的源，因此可以如下执行命令添加源：

sudo rpm -Uvh http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7.ngx.noarch.rpm

2、安装Nginx

　　通过yum search nginx看看是否已经添加源成功。如果成功则执行下列命令安装Nginx。

3、启动Nginx并设置开机自动运行

sudo systemctl start nginx.service

sudo systemctl enable nginx.service

****2. 安装并配置 Firekylin****

*安装 Firekylin*

在服务器上下载安装包

wget <https://firekylin.org/release/latest.tar.gz>

解压安装包

tar zvxf latest.tar.gz

安装程序依赖

cd firekylin

npm install

复制项目下的 pm2\_default.json 文件生成新文件 pm2.json

cp pm2\_default.json pm2.json

修改 pm2.json 文件中的 cwd 配置值为项目的当前路径 /root/firekylin：

{

"apps": [{

"name": "firekylin",

"script": "www/production.js",

"cwd": "/root/firekylin",

"exec\_mode": "fork",

"max\_memory\_restart": "1G",

"autorestart": true,

"node\_args": [],

"args": [],

"env": {

}

}]

}

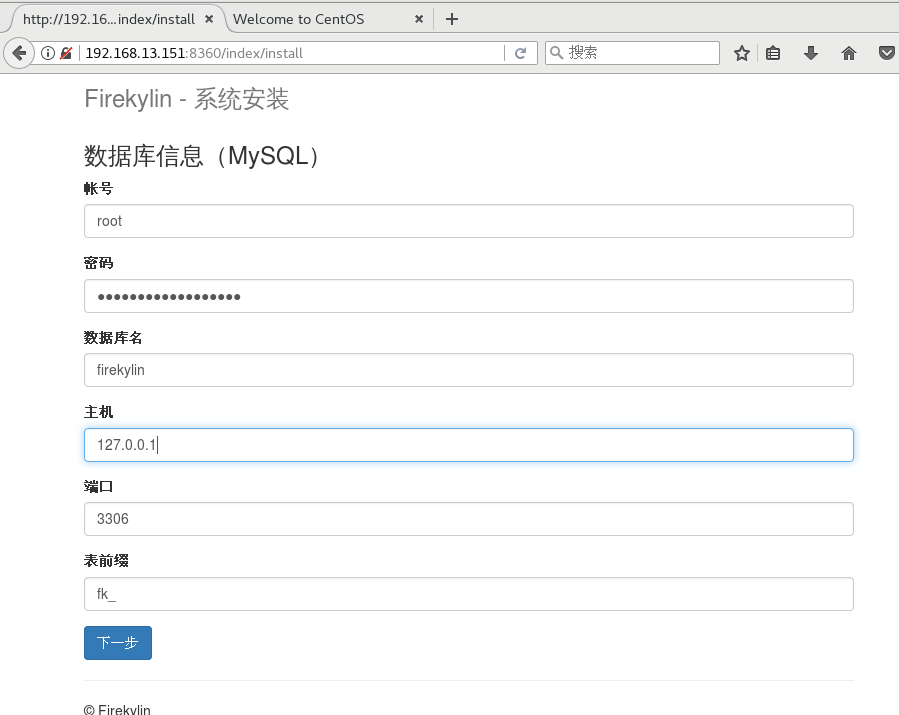
然后通过以下命令启动项目

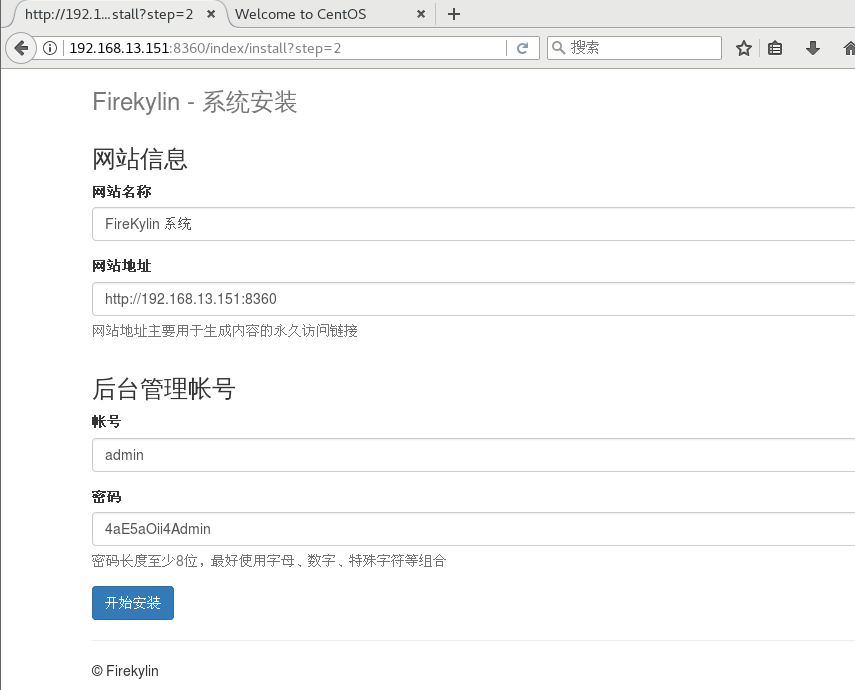
pm2 startOrReload pm2.json

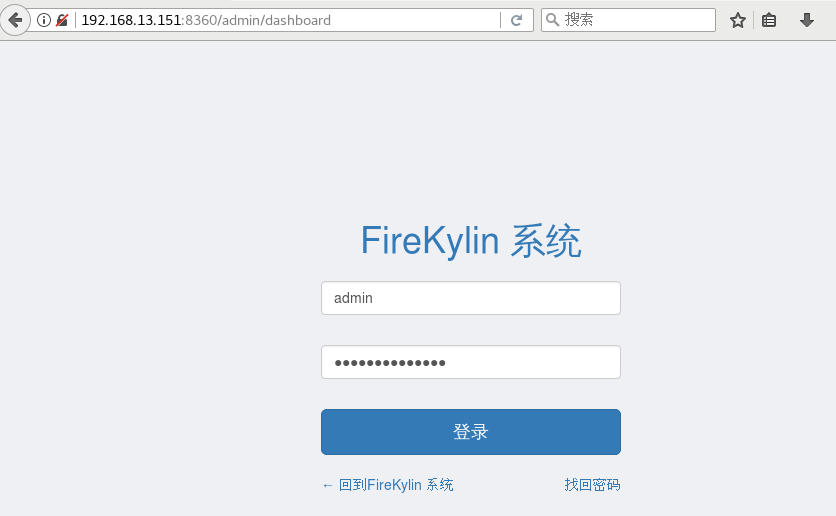
Firekylin 已经启动成功，使用浏览器直接访问 [http://yourip:8360/](http://yourip:8360/" \t "https://blog.csdn.net/qq_26954773/article/details/_blank) 或 [http://yourdomain.com:8360](http://yourdomain.com:8360/" \t "https://blog.csdn.net/qq_26954773/article/details/_blank)即可看到 Firekylin 的配置界面。

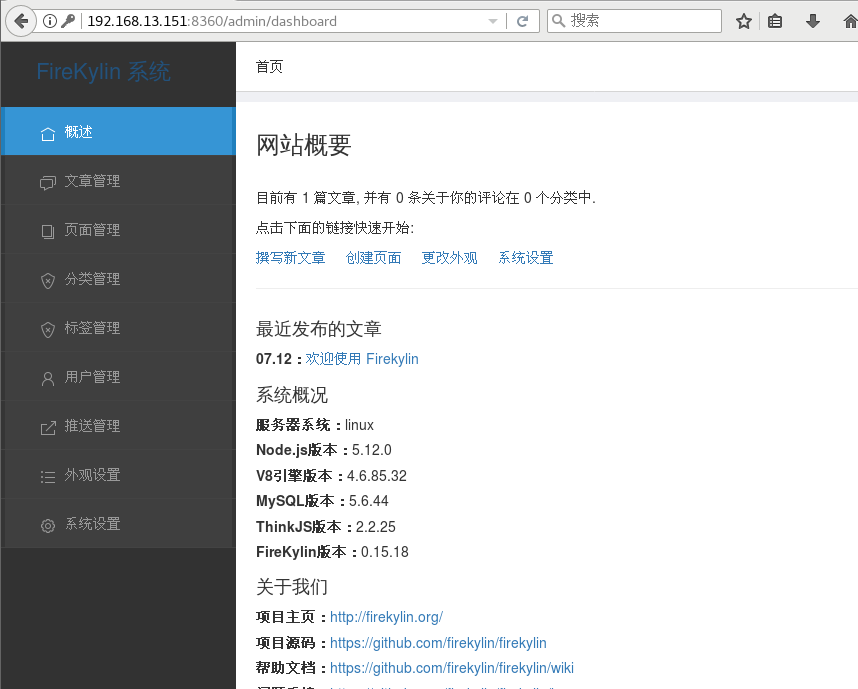
*配置信息*

配置过程输入参数如截图所示，其中数据库信息中的帐号字段设置为 root，密码字段设置为 4aE5aOii4Firekylin，数据库名字段设置为 firekylin，主机字段设置为 127.0.0.1，其他字段使用默认值；后台管理帐号中的帐号字段使用默认值 admin，密码字段设置为 4aE5aOii4Admin：



配置完成后可以通过后台管理帐号设置的帐号和密码登录博客管理后台，其值分别为 admin 和 4aE5aOii4Admin，截图如下所示：





*配置 Nginx*

下面我们就配置 Nginx 使用域名访问我们的网站了。   
复制项目下的 nginx\_default.conf 为 nginx.conf

cp nginx\_default.conf nginx.conf

修改 nginx.conf 文件：

server {

listen 80;

server\_name www.yourdomain.com; #将 www.yourdomain.com 替换为之前注册并解析的域名

root /root/firekylin;

set $node\_port 8360;

index index.js index.html index.htm;

location ^~ /.well-known/acme-challenge/ {

alias /root/firekylin/ssl/challenges/;

try\_files $uri = 404;

}

location / {

proxy\_http\_version 1.1;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header Host $http\_host;

proxy\_set\_header X-NginX-Proxy true;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection "upgrade";

proxy\_pass http://127.0.0.1:$node\_port$request\_uri;

proxy\_redirect off;

}

location = /development.js {

deny all;

}

location = /testing.js {

deny all;

}

location = /production.js {

deny all;

}

}

将 nginx.conf 文件软链到 nginx 配置目录下

service nginx restart

****3. 完成****