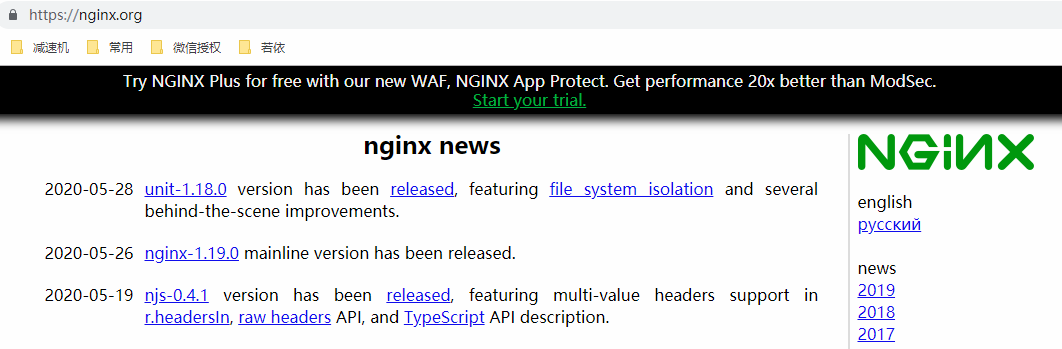
# Nginx 下载安装

## Nginx官网下载nginx

<https://nginx.org/>



下载windows 和linux版



## 安装nginx

上传服务器并解压

[agree@vm10-0-0-3 ~]$ tar -zxvf nginx-1.19.0.tar.gz

执行编译

[agree@vm10-0-0-3 nginx-1.19.0]$ ./configure

报错

./configure: error: the HTTP rewrite module requires the PCRE library.

You can either disable the module by using --without-http\_rewrite\_module

option, or install the PCRE library into the system, or build the PCRE library

statically from the source with nginx by using --with-pcre=<path> option.

[agree@vm10-0-0-3 nginx-1.19.0]$ yum list pcre-devel\*

Loaded plugins: fastestmirror, security

Loading mirror speeds from cached hostfile

Available Packages

pcre-devel.i686 7.8-7.el6 base

pcre-devel.x86\_64 7.8-7.el6 base

切换root用户安装

[root@vm10-0-0-3 ~]# yum install pcre-devel.x86\_64

Loaded plugins: fastestmirror, security

Loading mirror speeds from cached hostfile

Setting up Install Process

Resolving Dependencies

--> Running transaction check

---> Package pcre-devel.x86\_64 0:7.8-7.el6 will be installed

--> Processing Dependency: pcre = 7.8-7.el6 for package: pcre-devel-7.8-7.el6.x86\_64

--> Running transaction check

---> Package pcre.x86\_64 0:7.8-6.el6 will be updated

---> Package pcre.x86\_64 0:7.8-7.el6 will be an update

--> Finished Dependency Resolution

Dependencies Resolved

===================================================================================================================================================

Package Arch Version Repository Size

===================================================================================================================================================

Installing:

pcre-devel x86\_64 7.8-7.el6 base 320 k

Updating for dependencies:

pcre x86\_64 7.8-7.el6 base 196 k

Transaction Summary

===================================================================================================================================================

Install 1 Package(s)

Upgrade 1 Package(s)

Total download size: 516 k

Configuration summary

+ using system PCRE library

+ OpenSSL library is not used

+ using system zlib library

安装完成后再次执行

[agree@vm10-0-0-3 nginx-1.19.0]$ ./configure

nginx path prefix: "/usr/local/nginx"

nginx binary file: "/usr/local/nginx/sbin/nginx"

nginx modules path: "/usr/local/nginx/modules"

nginx configuration prefix: "/usr/local/nginx/conf"

nginx configuration file: "/usr/local/nginx/conf/nginx.conf"

nginx pid file: "/usr/local/nginx/logs/nginx.pid"

nginx error log file: "/usr/local/nginx/logs/error.log"

nginx http access log file: "/usr/local/nginx/logs/access.log"

nginx http client request body temporary files: "client\_body\_temp"

nginx http proxy temporary files: "proxy\_temp"

nginx http fastcgi temporary files: "fastcgi\_temp"

nginx http uwsgi temporary files: "uwsgi\_temp"

nginx http scgi temporary files: "scgi\_temp"

分别执行

[agree@vm10-0-0-3 nginx-1.19.0]$ make

[root@vm10-0-0-3 nginx-1.19.0]# make install

在默认安装目录启动nginx

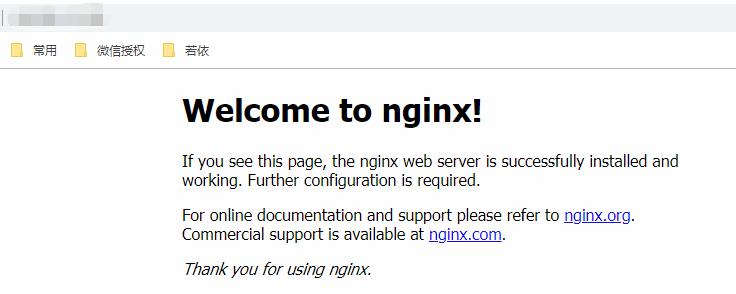
[root@vm10-0-0-3 sbin]# ./nginx

配置开机启动

vi /etc/rc.d/rc.local

/usr/local/nginx/sbin/nginx

直接输入地址连接成功



# Nginx配置

将nginx默认配置文件内容替换为以下内容，需要修改地址端口及项目资源路径

####main 全局配置块####

#配置运行Nginx服务器用户（组），解决13 permision权限问题 格式为 user root root;

user root;

#Nginx要开启的进程数。每个Nginx进程平均耗费10M~12M内存。建议指定和CPU的数量一致即可

worker\_processes 4;

#设置每个进程连接最大连接数

events {

worker\_connections 1024;

}

####http http配置块####

http {

#引入其他文件类型

include mime.types;

default\_type application/octet-stream;

#高效文件传输模式

sendfile on;

#连接保持超时时间

keepalive\_timeout 65;

####server server配置块####

server {

#服务监听端口

listen 80;

#用来指定IP地址或域名，多个域名之间用空格分开。

server\_name 120.92.151.50;

####location url 配置块####

#location支持正则表达式匹配，也支持条件判断匹配，用户可以通过location指令实现Nginx对动、静态网页进行过滤处理

location / {

#用于指定虚拟主机的网页根目录，这个目录可以是相对路径，也可以是绝对路径

root /home/ruoyi/dist;

try\_files $uri $uri/ /index.html;

#用于设定访问的默认首页地址

index index.html index.htm;

}

location /prod-api/{

proxy\_set\_header Host $http\_host;

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

proxy\_set\_header REMOTE-HOST $remote\_addr;

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

proxy\_pass http://120.92.151.50:8080/;

}

error\_page 500 502 503 504 /50x.html;

location = /50x.html {

root html;

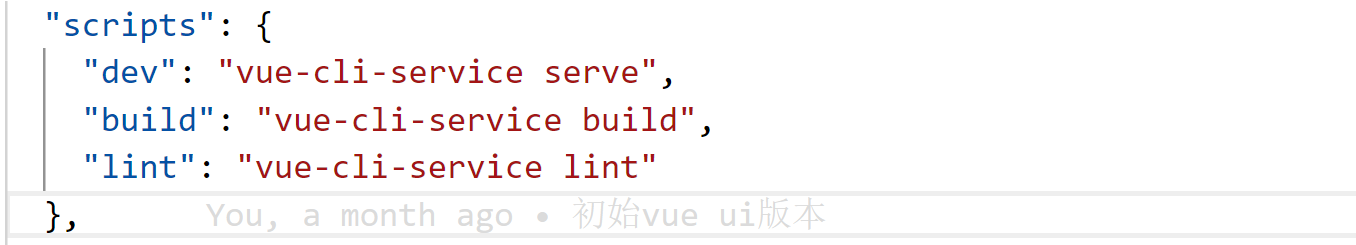
}

}

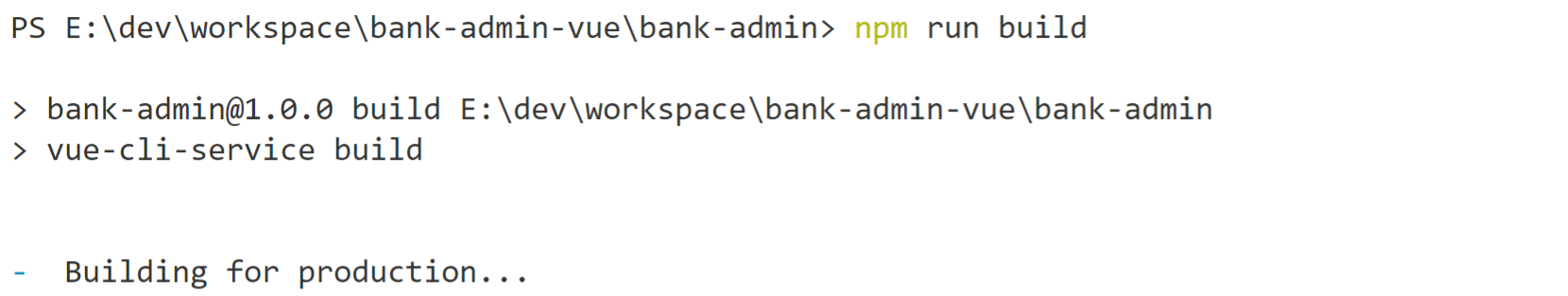
}

# 编译部署项目

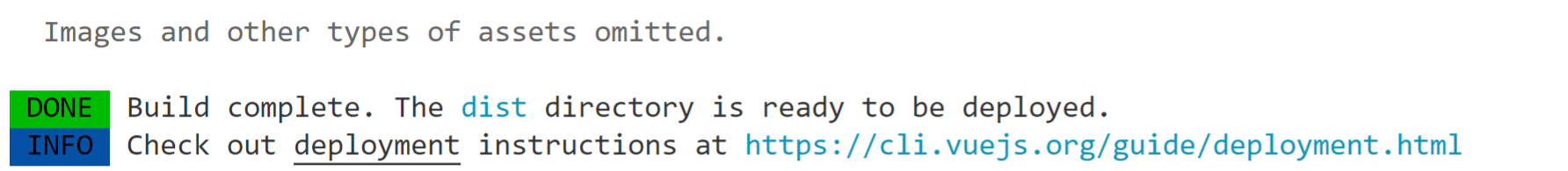
查看编译脚本命令伟build



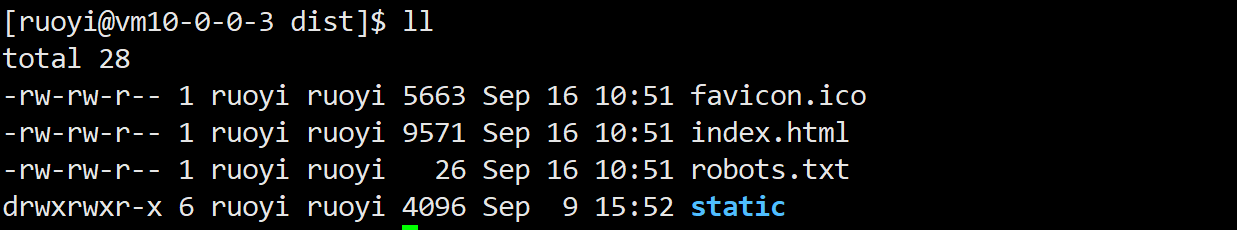
执行编译命令



编译完成后项目资源在dist文件夹中



上传项目资源dist至nginx指定的部署路径



在浏览器打开nginx配置的服务地址端口即可访问项目