可以先在服务器上安装好ngnix 和 PM2

Nginx –v 和 pm2 –v 查看版本号

这样就可以通过nginx端口映射来把前端的http请求转发到后端的nodejs项目上，nodejs项目本身是通过pm2来维护的

项目的部署(静态网站的后台website-server,电影网站的后台movie-server,微信小程序的后台indust-server，手机应用的后台gougou-server,以及微信公众号的后台wechat-server)



开源中国免费私有仓库：现在叫码云（上面东西不错哦）

在服务器是上拉下远端仓库的地址

先安装git sudo apt-get install git

Git 公钥

Cd .ssh

ls

Cd 当前目录下的 ~ .ssh文件夹， ls 查看id\_rsa.pub 公钥

Cat id\_rsa.pub

复制公钥，添加公钥到sshkey

通过git 就可以把本地的的代码上传到git仓库，服务器也可以拉取到远端的git代码，这样就可以实现三端代码的打通了

一、静态网站的部署

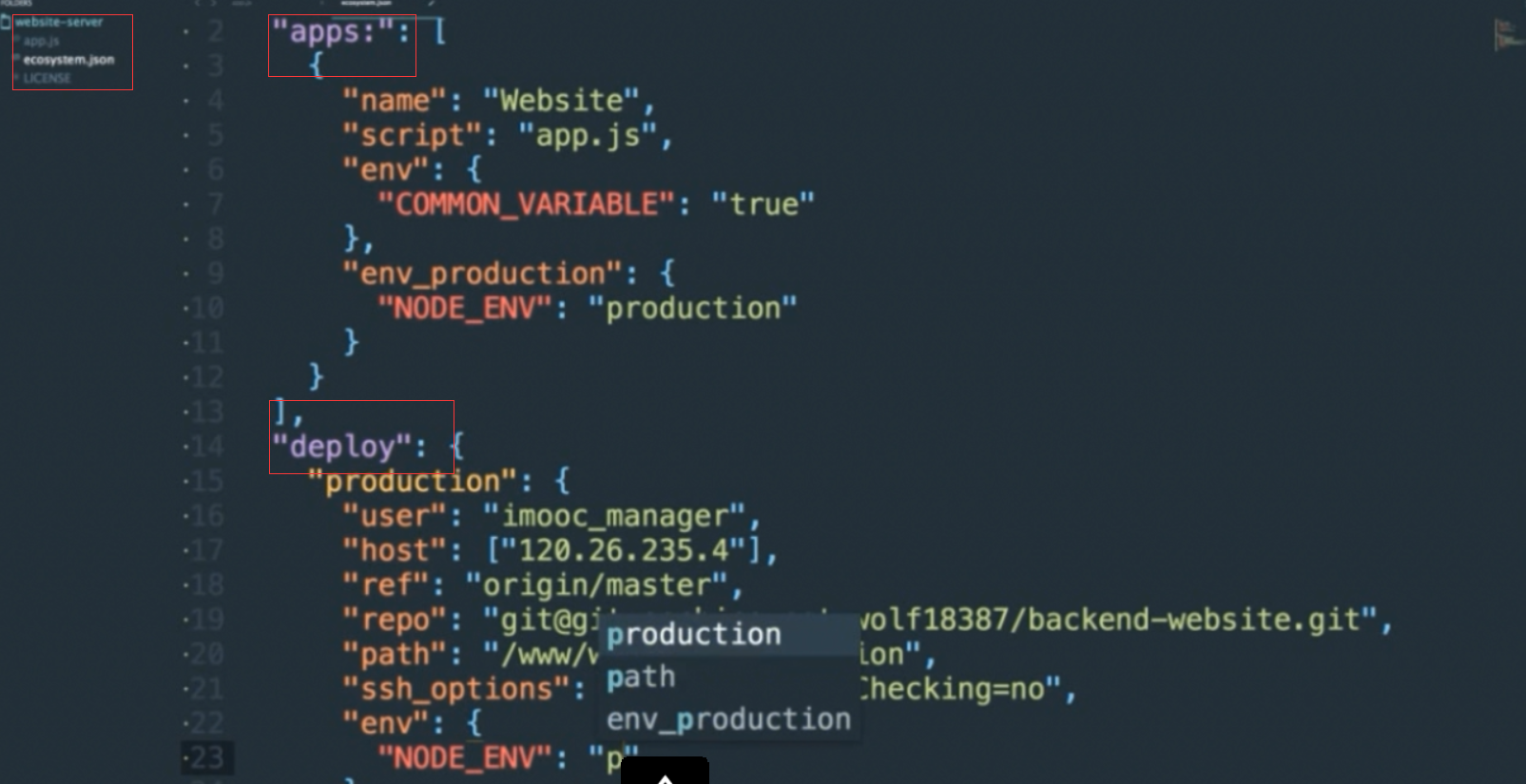
通过PM2来傻瓜式部署nodejs代码，

通过查看官方文档pm2.keymetrics.io

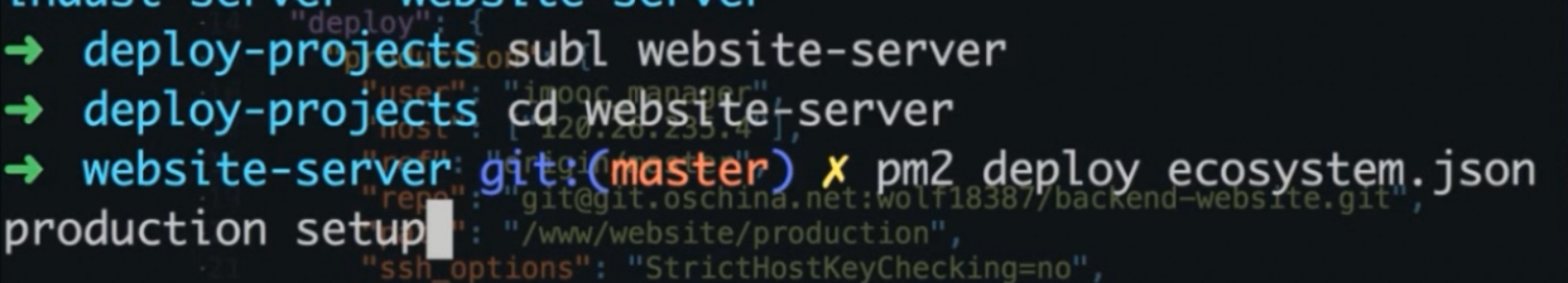
从部署到日志，一直到配置参数，等等都很详细

通过配置文件来部署，

项目更目录新建一个ecosystem.json



配置完成后 在本地通过执行命令pm2 deploy ecosystem.json



进入到服务器终端，创建www/website文件夹，可能部署失败，这可能是是website文件夹对imooc——manager这个没有操作权限的问题

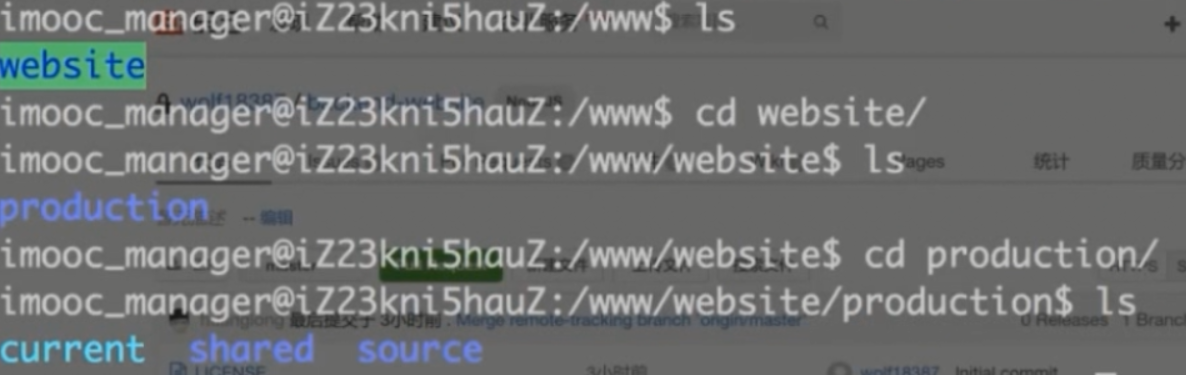
Ls –all 来查看文件夹权限

修改website 文件夹权限



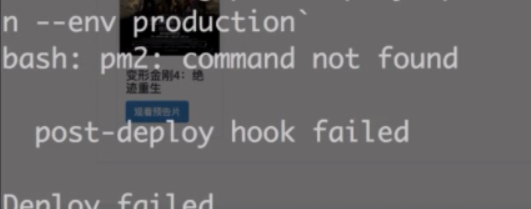
最好是给imooc\_manage添加到具有这些权限的组里面，而不是直接操作这些文件的权限

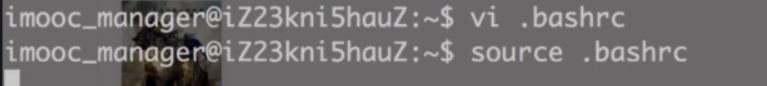
修改权限成功后，可以在本地执行pm2 的部署命令了。



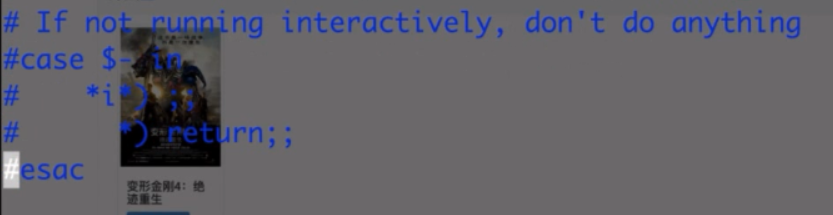
这时候服务器的带代码就已经准备就绪了，只需在本地客户端命令行中部署代码

一般会部署失败，

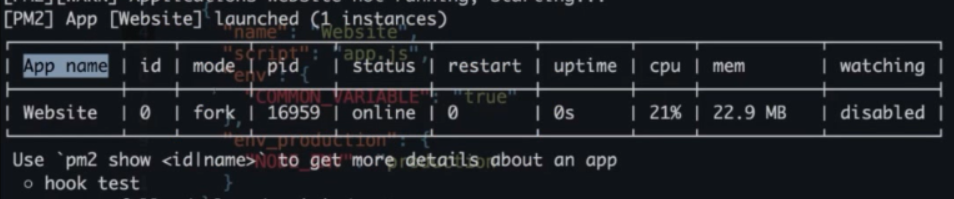
这是因为我们的pm2命令是非交互的，在服务器端的根目录下，编辑bashrc文件



注释这几行

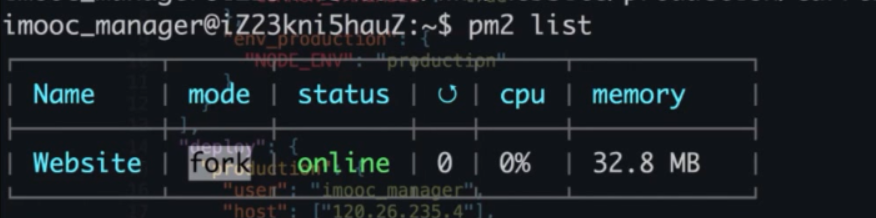


重新本地发布下



这就是部署成功后的表格

此时在服务器端也可以通过 pm2 list 来查看发布的应用





Pm2 部署代码的原理：通过本地的pm2, 在本地的通过命令行，登陆服务器，通知服务器clone git 仓库的代码，部署到相应的文件夹中，这是就可以发布（部署）我们的项目了

首先是部署静态站点，这不和数据库打交道，如果是有数据库，这必须在服务器上配置好数据，并能跑起来，并且要管理和数据库用户的权限和密码一些操作。

项目是发布了，但是不能确保项目在线上就能正常访问到

还**要做两件事**

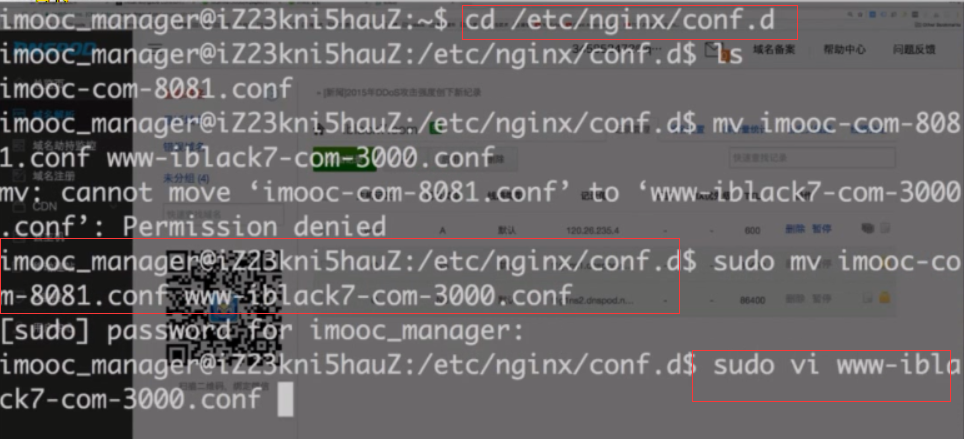
1. **修改域名指向**

可以在pod域名管理中将域名配置到对应服务器的ip地址

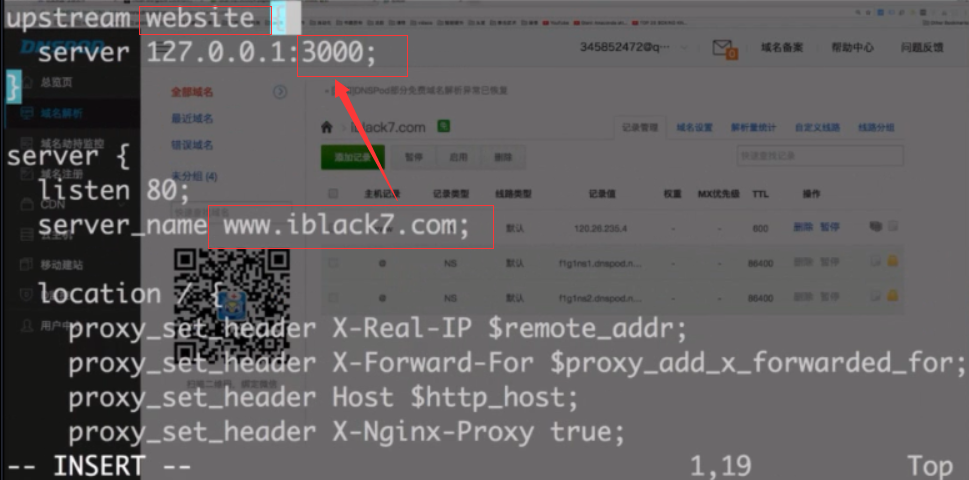


1. **修改Ngnix配置，让nginx识别这个域名后，将请求转发到我们的app.js中的3000端口**

找到文件，编辑



编辑nginx



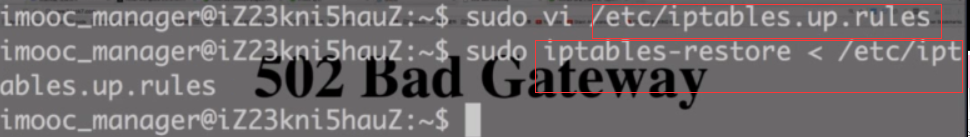
Nginx将[www.iblack7.com下80端口的请求转发到(服务器)本机的3000](http://www.iblack7.com下80端口的请求转发到(服务器)本机的3000)端口

保存退出后，重启nginx



502 bad getway

所以要允许开启允许3000端口的访问权限





**这样就可以访问了，如果项目要改动，怎么在线上反应出来呢？需要三步：**

1. **修改项目代码**
2. **将新的项目代码提交到git远程仓库**
3. **通过pm2 重新部署就可以了**

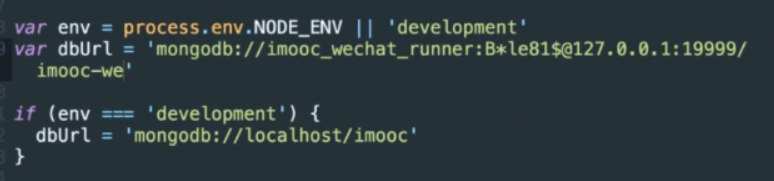
二、电影网站的部署

特点：有一些静态资源，并且链接了mongodb 数据库

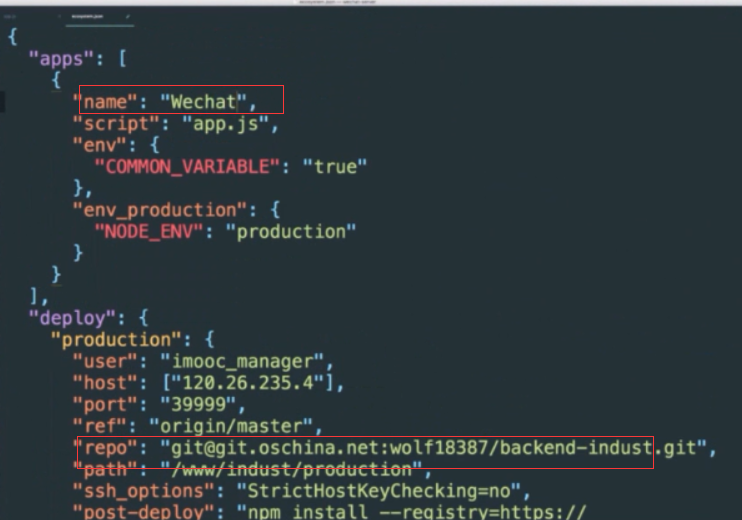
1. 在dnspod网站域名管理中配置域名，这样就可以通过movie.iblack7.com来访问到我们的电影网站内容

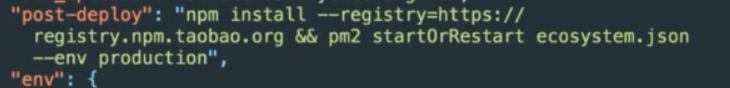


因为有数据库连接，所以在代码入口文件中中兼容数据库源



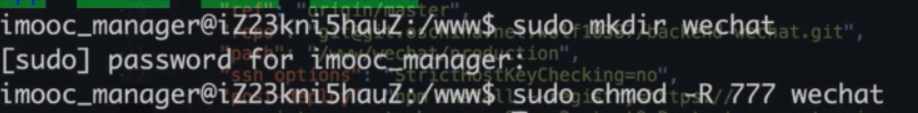
1. 在项目根目录下创建pm2的配置文件ecosystem.json 文件，修改一些配置项



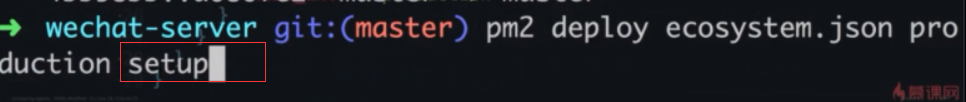


编辑post-deploy的钩子

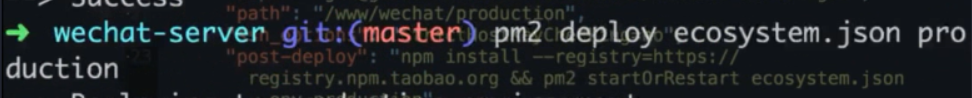
1. 在服务器的发布目录下，创建发布目录，修改发布目录权限



1. 将本地代码提交到git仓库，通过pm2 部署项目，并发布



发布



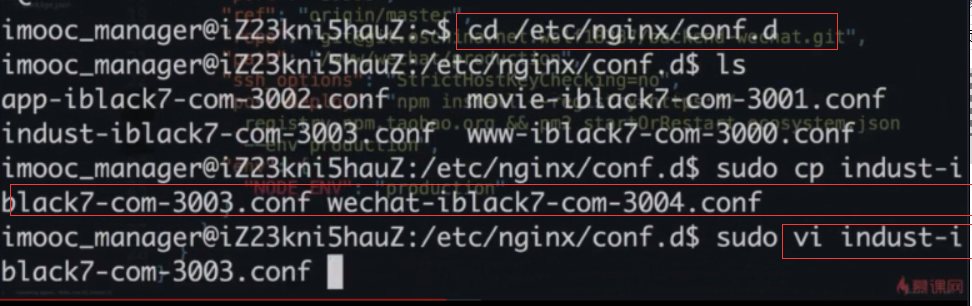
在服务器上检查发布应用的一些发布状态



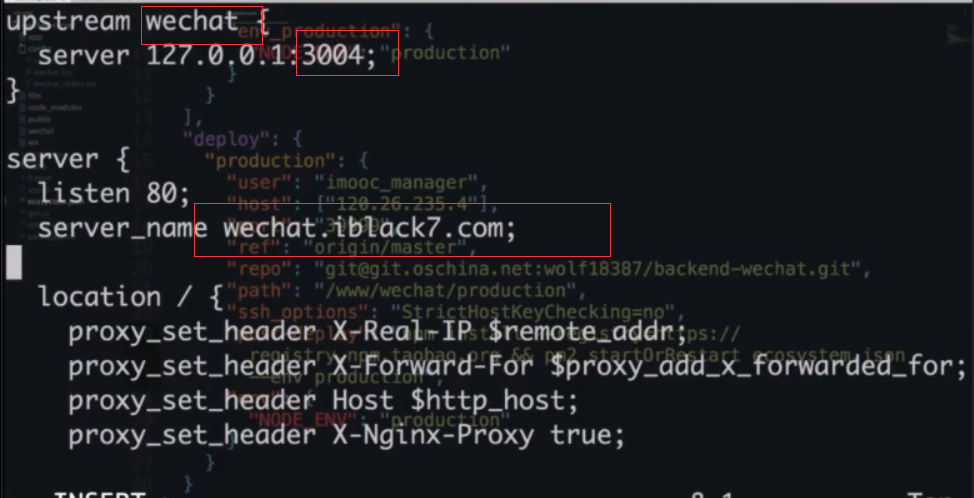
查看日志 pm2 logs 是否有日志报错

1. 配置nginx，转发请求

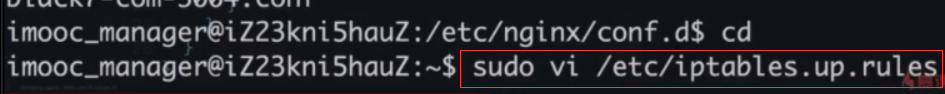
打开 拷贝，编辑

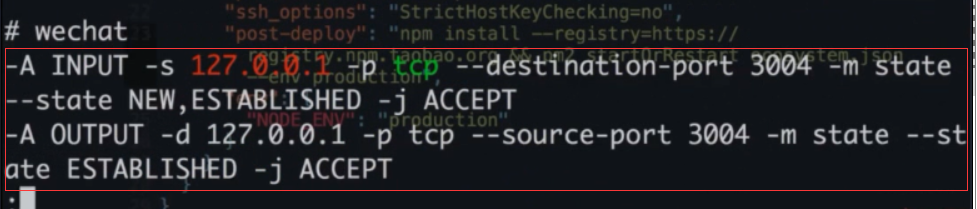


修改，保存

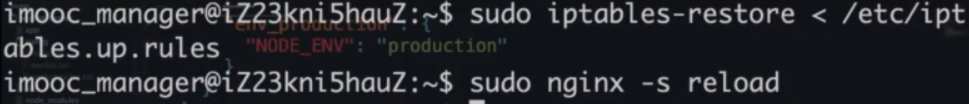


1. 修改防火墙，新增3004可以被访问的防火墙





配置完后，重新载入iptables,重启nginx



1. 重新发布



基本实现了一个项目的部署是什么样了

其他几个都是类似的流程