1.简介安装

Portus,Portus和harbor都是开源的Registry服务器，官方站点`http://port.us.org/`

1.克隆git

```
[root@linuxea.com /data]# git clone https://github.com/SUSE/Portus.git
正克隆到 'Portus'...
remote: Counting objects: 18528, done.
remote: Total 18528 (delta 0), reused 0 (delta 0), pack-reused 18528
接收对象中: 100% (18528/18528), 21.80 MiB | 104.00 KiB/s, done.
```

国内下载较慢，我们可以先下载镜像，在docker-compose中，分别有library/mariadb:10.0.23，library/registry:2.3.1，我们可以先手动pull
2.开始安装

```
[root@linuxea.com /data/Portus]# ./compose-setup.sh -f -e 10.57.57.57

############
# WARNING #
############

This deployment method is intended for testing/development purposes.
To deploy Portus on production please take a look at: http://port.us.org/documentation.html

The setup will destroy the containers used by Portus, removing also their volumes.
No stopped containers
Creating portus_db_1
Creating portus_crono_1
Creating portus_web_1
Creating portus_registry_1
Waiting for mariadb to be ready in 5 seconds
Portus: configuring database... failed, will retry
Waiting for mariadb to be ready in 5 seconds
Portus: configuring database... [SUCCESS]

####################
#     SUCCESS     #
####################

Make sure port 3000 and 5000 are open on host 10.57.57.57

Open http://10.57.57.57:3000 with your browser and perform the following steps:

  1. Create an admin account
  2. You will be redirected to a page where you have to register the registry. In this form:
    - Choose a custom name for the registry.
    - Enter 10.57.57.57:5000 as the hostname.
    - Do *not* check the "Use SSL" checkbox, since this setup is not using SSL.

Perform the following actions on the docker hosts that need to interact with your registry:

- Ensure the docker daemon is started with the '--insecure-registry 10.57.57.57:5000'
- Perform the docker login.

To authenticate against your registry using the docker cli do:

  $ docker login -u <portus username> -p <password> -e <email> 10.57.57.57:5000

To push an image to the private registry:

  $ docker pull busybox
  $ docker tag busybox 10.57.57.57:5000/<username>/busybox
  $ docker push 10.57.57.57:5000/<username>/busybox
[root@linuxea.com /data/Portus]#
```

3.添加iptables

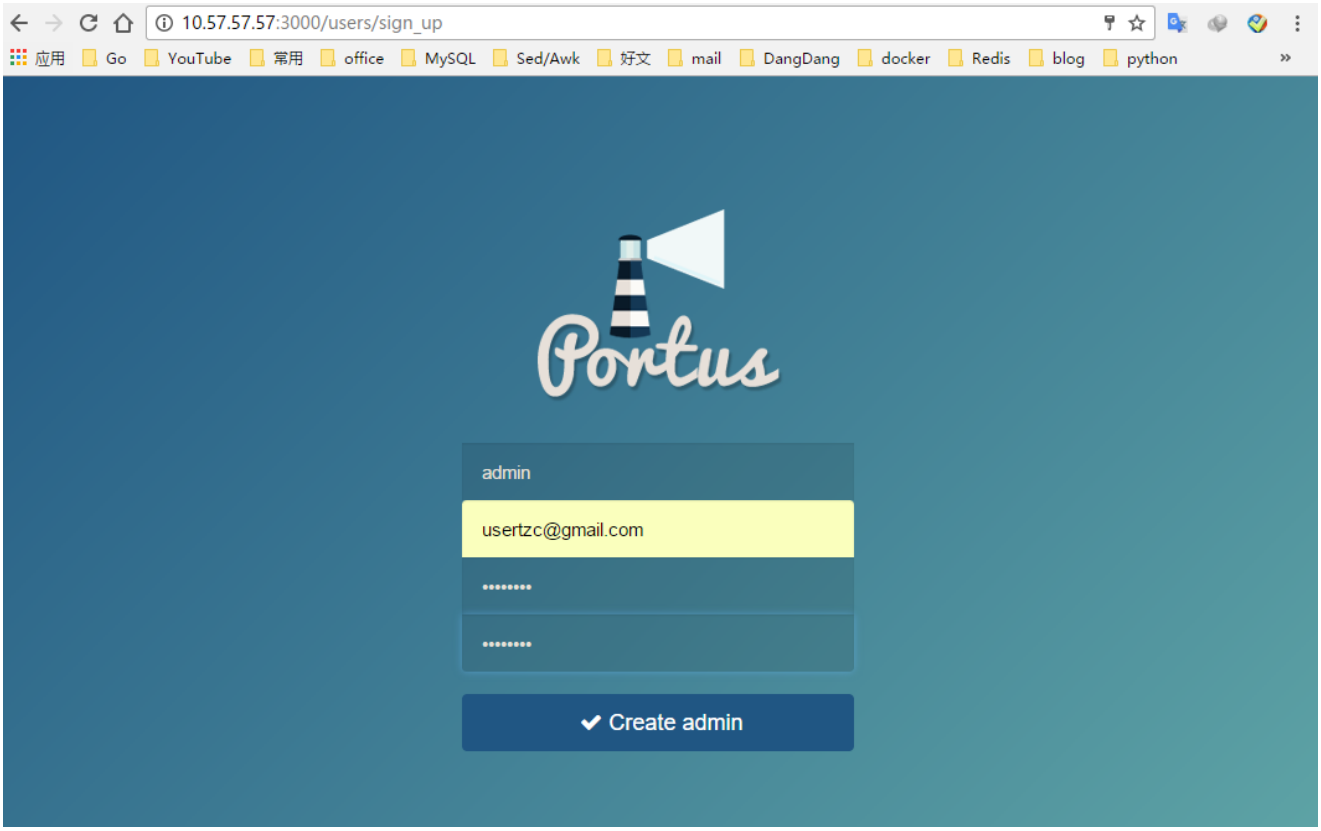
```
[root@linuxea.com /data/Portus]# iptables -I INPUT 5  -p tcp -m state --state NEW -m tcp -m
multiport --dports 3000,5000 -m comment --comment "Portus" -j ACCEPT
```

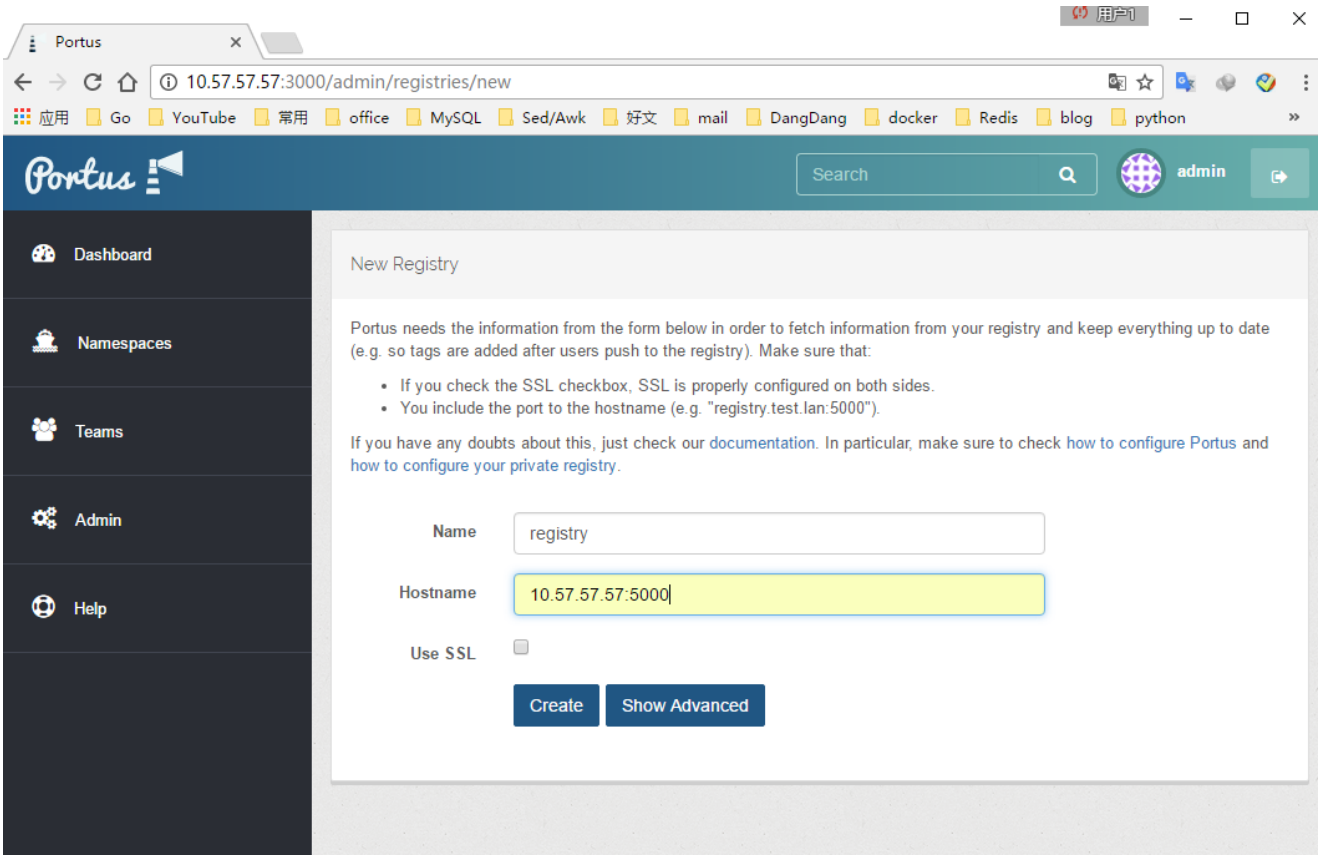
可以看到已经启动了4个容器

```
[root@linuxea.com /data/Portus]# docker ps -a
CONTAINER ID        IMAGE                     COMMAND                  CREATED             STATUS
PORTS                                NAMES
181713ebf12a        library/registry:2.3.1    "/bin/registry /etc/d"   31 minutes ago      Up 3
minutes        0.0.0.0:5000-5001->5000-5001/tcp    portus_registry_1
4613cb2b2ee7        portus_web                "bin/crono"              31 minutes ago      Up 3
minutes        3000/tcp                            portus_crono_1
c8bdba50f542        portus_web                "puma -b tcp://0.0.0."   31 minutes ago      Up 3
minutes        0.0.0.0:3000->3000/tcp              portus_web_1
0a989947a943        library/mariadb:10.0.23   "/docker-entrypoint.s"   31 minutes ago      Up 3
minutes        3306/tcp                            portus_db_1
```

2.创建管理员账户

1.打开ip:port，默认第一个注册的则是管理员

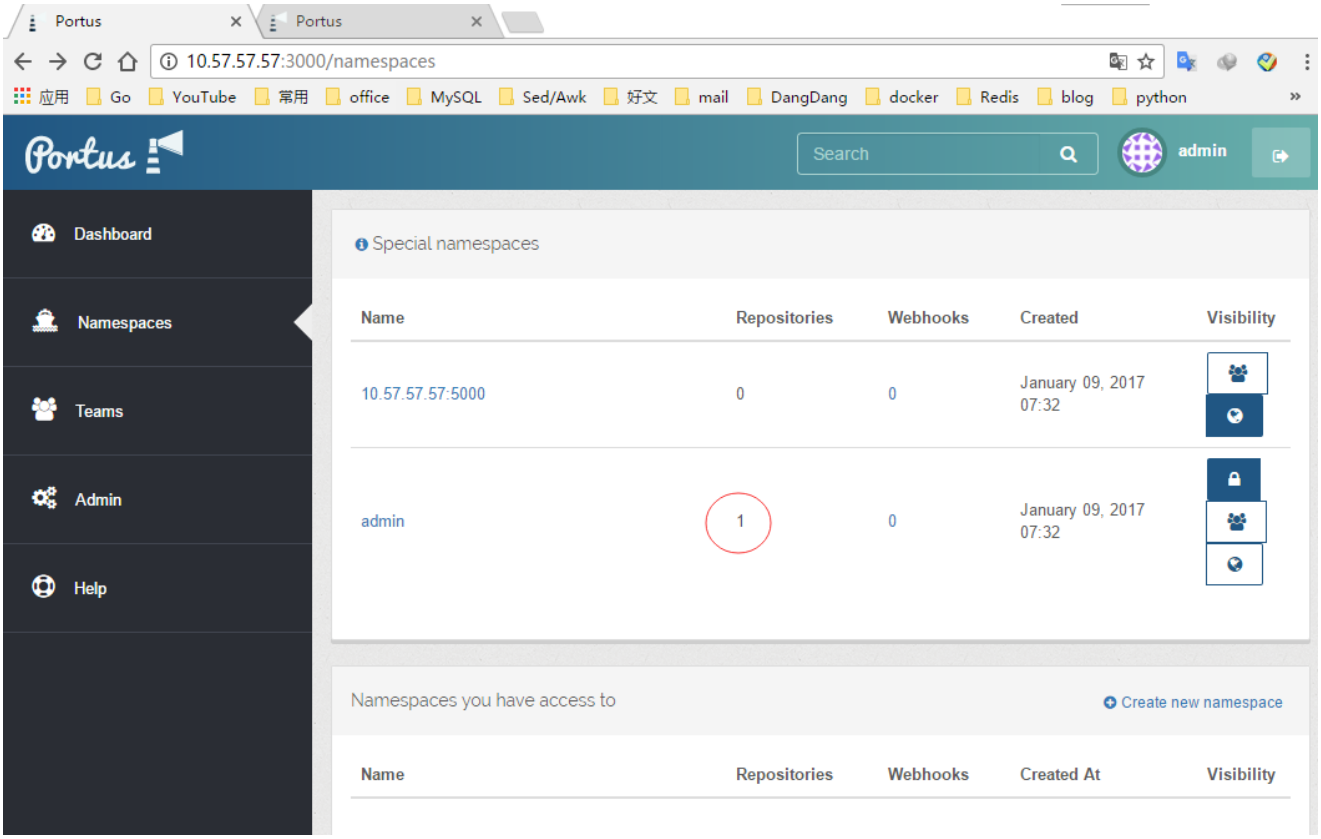




3.上传镜像

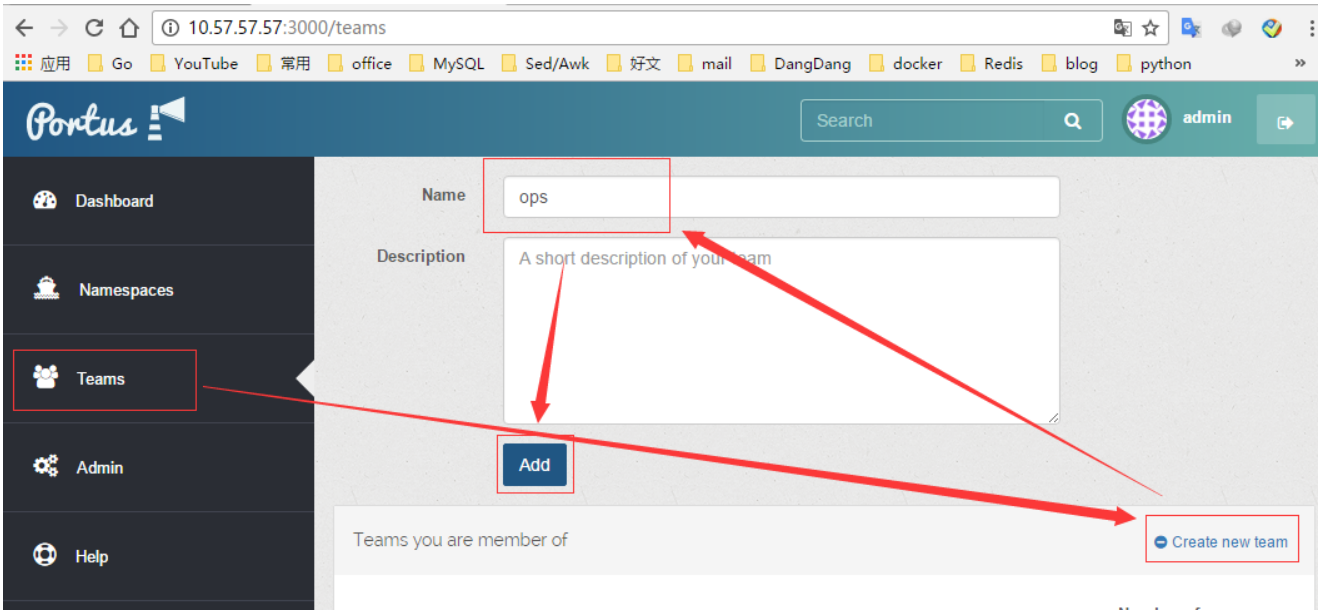
1.push镜像到admin用户下

```
[root@linuxea.com /data/Portus]# docker tag alpine 10.57.57.57:5000/admin/alpine
[root@linuxea.com /data/Portus]# docker push 10.57.57.57:5000/admin/alpine
The push refers to a repository [10.57.57.57:5000/admin/alpine]
7cbcbac42c44: Pushed
latest: digest: sha256:a4104316f43c73146f1c0af4747d88047a808e58238bcad6506a7fbbf3b30b90 size: 528
[root@linuxea.com /data/Portus]#
```

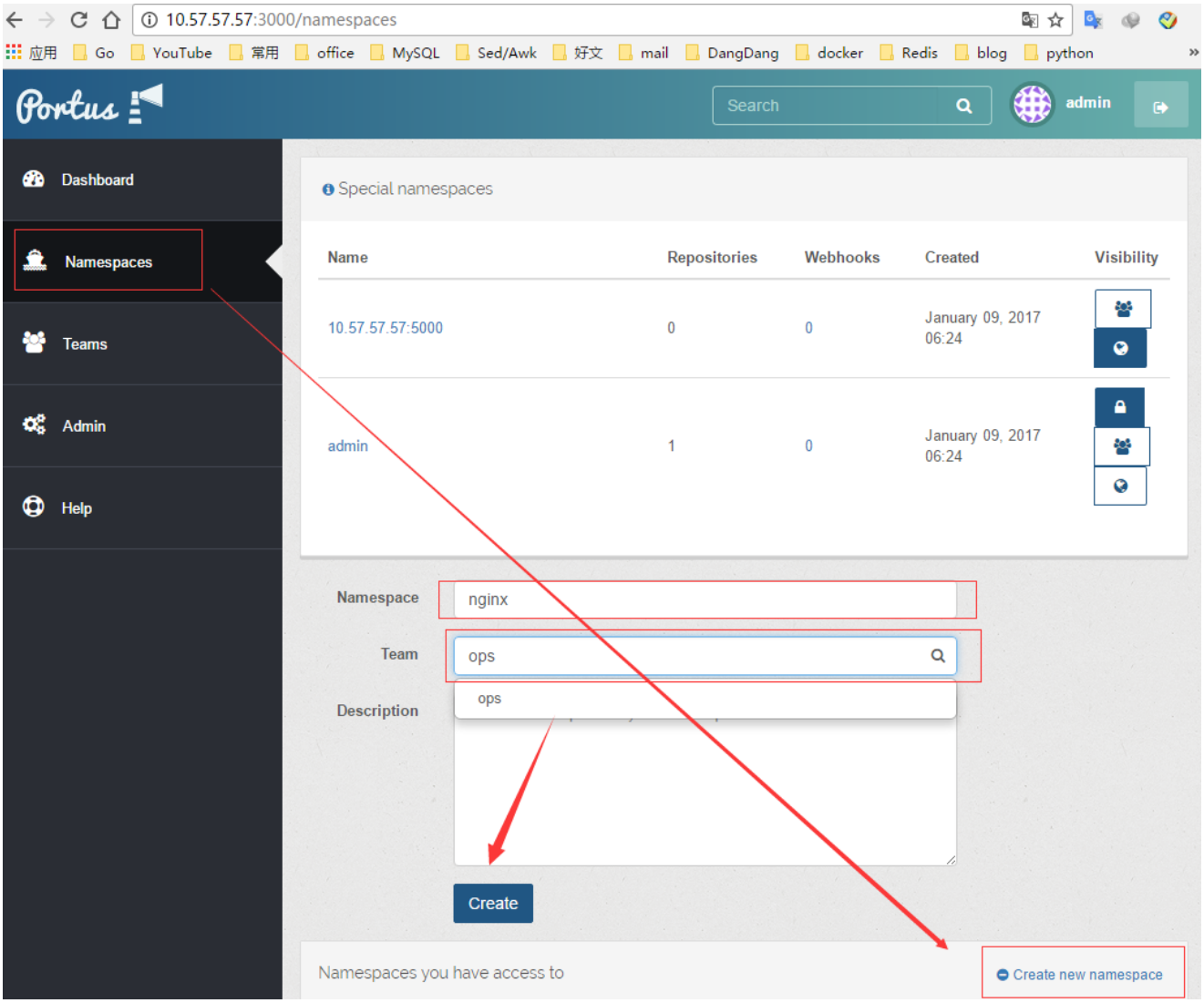


4.创建组

创建teams，加入到namespaces



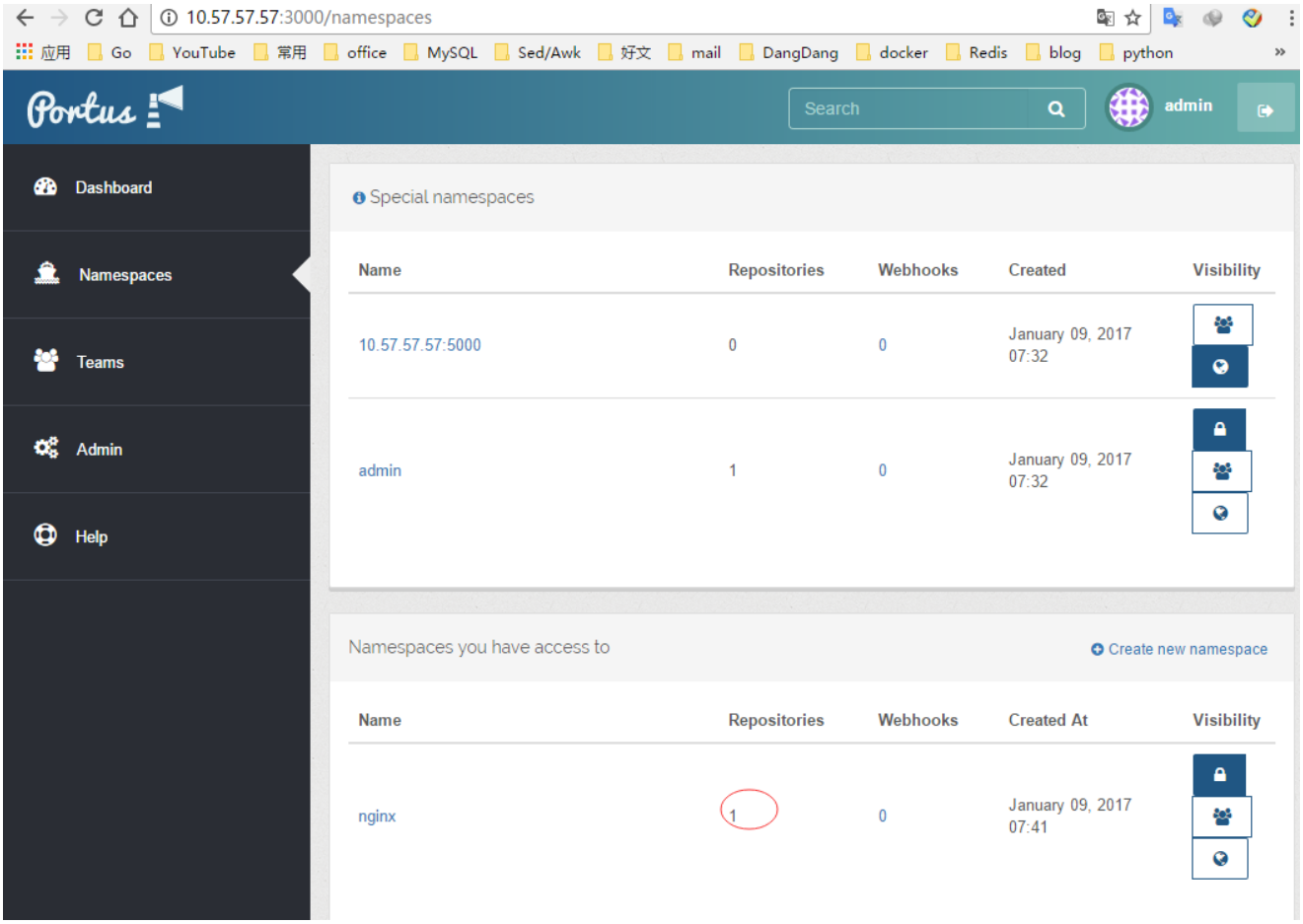
创建ops团队后，创建一个nginx的namespaces,也就是项目，将ops加入到namaspaces中。



我们上传一个nginx镜像到nginx项目下

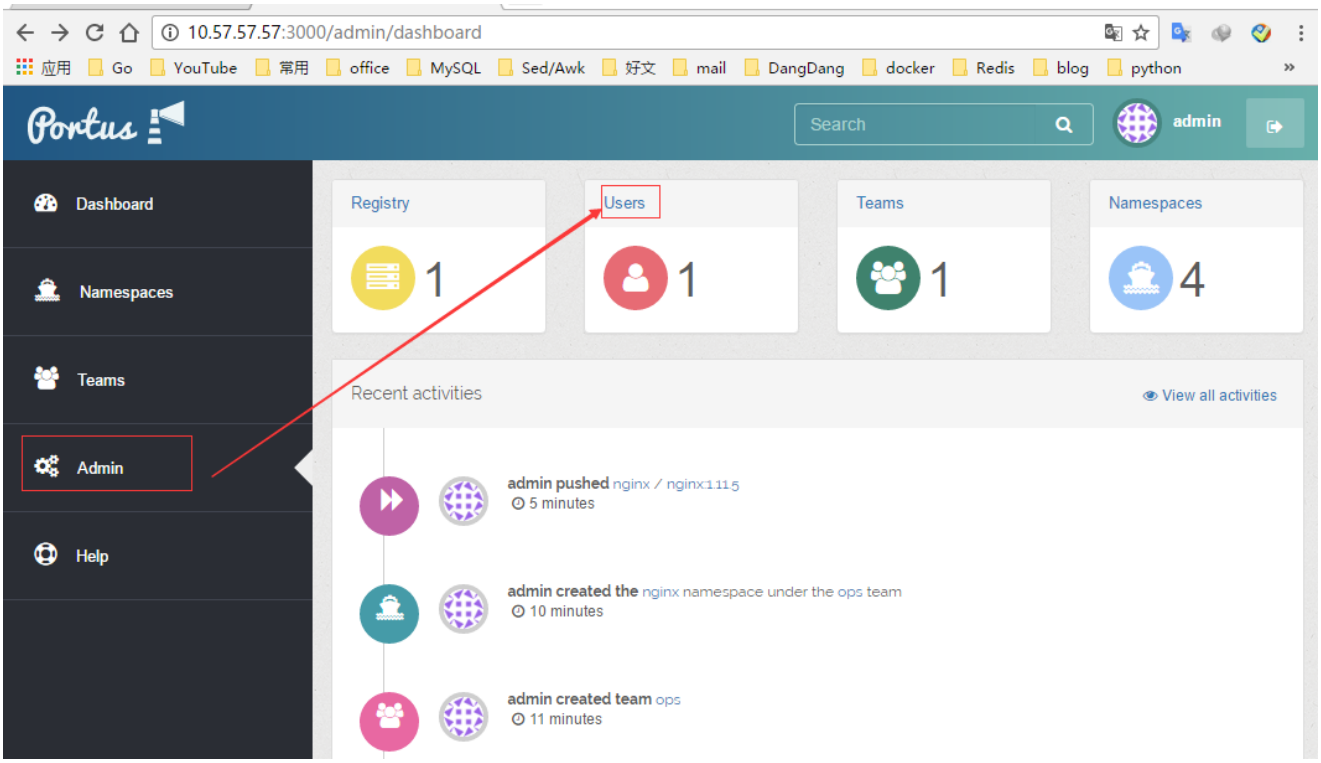
```
[root@linuxea.com /data/Portus]# docker tag nginx:1.11.5 10.57.57.57:5000/nginx/nginx:1.11.5
[root@linuxea.com /data/Portus]# docker push 10.57.57.57:5000/nginx/nginx:1.11.5
The push refers to a repository [10.57.57.57:5000/nginx/nginx]
3f117c44afbb: Pushed
c4a8b7411af4: Pushed
fe4c16cbf7a4: Pushed
1.11.5: digest: sha256:9038d5645fa5fcca445d12e1b8979c87f46ca42cfb17beb1e5e093785991a639 size: 948
```

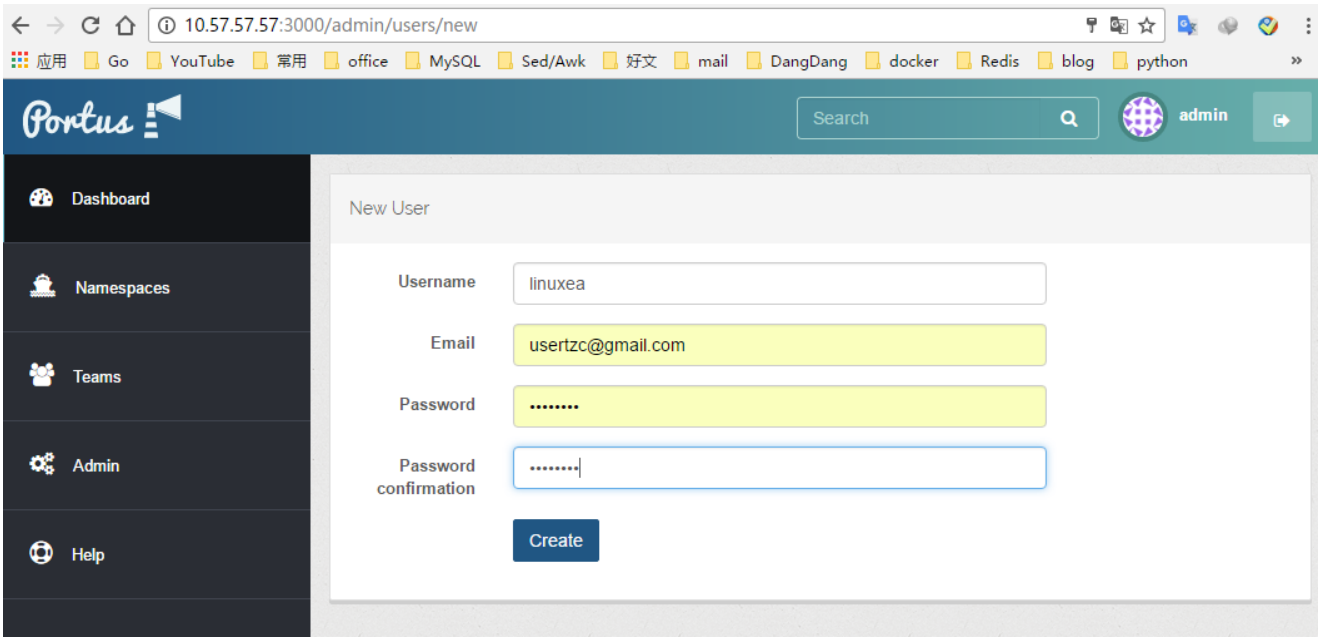
已经上传至



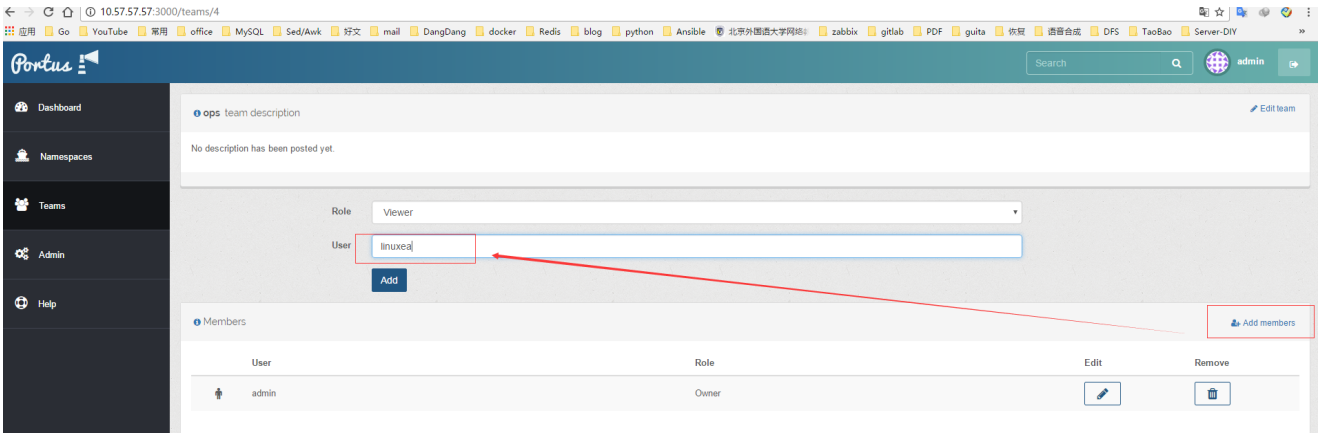
5.验证create普通用户权限

1，创建用户





2，把linuxea添加到nginx中



3，测试普通用户权限：

```
[root@linuxea.com ~]# echo '{ "insecure-registries":["10.57.57.57:5000"] }' >
/etc/docker/daemon.json
[root@linuxea.com ~]# systemctl restart docker
[root@linuxea.com ~]# docker login -u linuxea  -p 12345678 10.57.57.57:5000
Login Succeeded
[root@linuxea.com ~]# docker pull 10.57.57.57:5000/nginx/nginx:1.11.5
1.11.5: Pulling from nginx/nginx
386a066cd84a: Pull complete
7bdb4b002d7f: Pull complete
49b006ddea70: Pull complete
Digest: sha256:9038d5645fa5fcca445d12e1b8979c87f46ca42cfb17beb1e5e093785991a639
Status: Downloaded newer image for 10.57.57.57:5000/nginx/nginx:1.11.5
```

4，普通用户尝试上传unauthorized: authentication required

```
[root@linuxea.com ~]# docker tag mariadb:10.0.23 10.57.57.57:5000/nginx/mariadb:10.0.23
[root@linuxea.com ~]# docker push 10.57.57.57:5000/nginx/mariadb:10.0.23
The push refers to a repository [10.57.57.57:5000/nginx/mariadb]
5f70bf18a086: Preparing
5f70bf18a086: Preparing
978a35efaa8c: Preparing
494fda86ea11: Preparing
577a7512a96d: Preparing
1f5651eebf56: Waiting
38b6325acdbb: Waiting
1fd270dc6ea8: Waiting
c2bf450a065b: Waiting
ddef881aeaff: Waiting
unauthorized: authentication required
```