

1.环境装备:

centos7.2.1511,python2.7以上版本, docker引擎1.10或者更高版本, docker compose需要1.6.0或者更高版本

harbor安装参考: https://github.com/vmware/harbor/blob/master/docs/installation_guide.md

harbor下载地址: <https://github.com/vmware/harbor/releases>

vmware安装参考: https://github.com/vmware/harbor/blob/master/docs/installation_guide_ova.md

安装包分为离线, 网络, 源码安装

网络安装包: <https://github.com/vmware/harbor/releases/download/0.5.0/harbor-online-installer-0.5.0.tgz>

离线安装包: <https://github.com/vmware/harbor/releases/download/0.5.0/harbor-offline-installer-0.5.0.tgz>

源代码: <https://github.com/vmware/harbor/archive/0.5.0.tar.gz>

2.配置参数:

在harbor.cfg中

修改hostname = 域名

配置参数位于文件harbor.cfg。参数说明如下-请注意, 最起码, 你需要改变主机属性。

主机: 目标主机的主机名, 这是用于访问用户界面和注册表服务。它应该是IP地址或目标计算机的完全限定域名(FQDN), 例如, 192.168.1.10或reg.yourdomain.com。不要使用localhost或127.0.0.1作为主机名-注册表服务需要由外部客户端访问!

ui_url_protocol: (HTTP或HTTPS, 默认为HTTP) 用于访问用户界面和令牌/通知服务的协议。默认情况下, 这是HTTP。要建立HTTPS协议, 请参阅配置与港HTTPS访问。

电子邮件设置: 需要对港这些参数, 以便能够向用户发送一个“密码重设”电子邮件, 以及是否需要该功能只是必要的。还有, 千万注意, 在默认情况下SSL连接是没有启用-如果你的SMTP服务器需要SSL, 但不支持STARTTLS, 那么你应该通过设置启用SSL email_ssl = TRUE。

email_server = smtp.mydomain.com

email_server_port = 25

email_username = sample_admin@mydomain.com

email_password = abc

EMAIL_FROM = 管理员sample_admin@mydomain.com

email_ssl = false

harbor_admin_password: 管理员的初始密码。此密码仅在港口首次发布时生效。之后, 将忽略此设置, 并且应在UI中设置管理员的密码。需要注意的是默认的用户名/密码管理/ Harbor12345。

auth_mode: 用于身份验证的类型。缺省情况下, 它是db_auth, 即凭证存储在数据库中。对于LDAP身份验证, 此设置了ldap_auth。

ldap_url: 本LDAP端点URL (例如ldaps://ldap.mydomain.com)。仅当使用auth_mode设为了ldap_auth。

ldap_searchdn: 谁有权搜索LDAP / AD服务器 (例如权限的用户的DN

uid=admin,ou=people,dc=mydomain,dc=com)。

ldap_search_pwd: 由指定的用户的密码ldap_searchdn。

LDAP_BASEDN: 基本DN来查找用户, 例如ou=people,dc=mydomain,dc=com。仅当使用auth_mode设为了ldap_auth。

ldap_filter: 用于查找用户, 例如搜索筛选器(objectClass=person)。

LDAP_UID: 用于一个LDAP搜索期间用户匹配的属性, 也可以是流体, CN, 电子邮件或其他属性。

ldap_scope: 范围以搜索用户, 1- LDAP_SCOPE_BASE, 2- LDAP_SCOPE_ONELEVEL, 3- LDAP_SCOPE_SUBTREE。默认值为3。

DB_PASSWORD: 对于用于MySQL数据库的root密码db_auth。更改此密码以用于任何生产使用!

self_registration:（上或关闭，默认为上）启用/禁用用户注册自己的能力。禁用时，新用户只能由管理员用户创建，只有管理员用户才能在Harbor中创建新用户。注：当auth_mode设为了ldap_auth，自注册功能总是禁用，这个标志将被忽略。

use_compressed_js:（上或关闭，默认为上），供生产使用，把这个标志上。在开发模式下，将其设置为关闭，这样的JS文件可以单独进行修改。

max_job_workers:（默认值为3）复制工人就业服务的最大数量。对于每个图像复制作业，工作程序将存储库的所有标记同步到远程目标。增加此数目允许系统中更多的并发复制作业。但是，由于每个工人消耗一定量的网络/CPU / IO资源，请根据主机的硬件资源仔细选择此属性的值。

token_expiration: 到期时间（分钟）通过令牌服务创建的令牌，默认为30分钟。

verify_remote_cert:（上或关闭，默认为上）该标志决定是否验证SSL / TLS证书时，港口，远程注册表实例通信。设置这个属性关绕过SSL / TLS验证，这是通常用在远程实例有自签名或不受信任的证书。

customize crt:（上或关闭，默认为上），如果此属性上，在准备脚本创建注册表的令牌生成/验证私钥和根证书。以下属性：crt_country, crt_state, crt_location, crt_organization, crt_organizationalunit, crt_commonname, crt_email被用作产生密钥的参数。该属性设置为关闭时，密钥和根证书是由外部来源提供。请参阅自定义键和港令牌服务证书获取更多信息。

3.开始安装

```
[root@linuxea.com ~]# yum install openssl openssl-devel
[root@linuxea.com ~]# mkdir -p /data/cert && cd /data/cert/
[root@linuxea.com /data/cert/]# openssl req -newkey rsa:4096 -nodes -sha256 -keyout ca.key -x509 -days 365 -out ca.crt -subj "/C=CN/L=London/O=Company Ltd/CN=nginx-docker"
[root@linuxea.com /data/cert/]# openssl req -newkey rsa:4096 -nodes -sha256 -keyout linuxea.reg.com.key -out server.csr -subj "/C=CN/L=London/O=Company Ltd/CN=linuxea.reg.com"
[root@linuxea.com /data/cert/]# openssl x509 -req -days 365 -in server.csr -CA ca.crt -CAkey ca.key -CAcreateserial -out linuxea.reg.com.crt
[root@linuxea.com /data/cert/]# cd ..
```

download harbor5.0

harbor git地址: <https://github.com/vmware/harbor/releases>

离线安装包

```
[root@linuxea.com /data/]# wget -c
https://github.com/vmware/harbor/releases/download/0.5.0/harbor-offline-installer-0.5.0.tgz
```

在线安装包

```
[root@linuxea.com /data/]# wget -c
https://github.com/vmware/harbor/releases/download/0.5.0/harbor-online-installer-0.5.0.tgz
```

修改配置文件

```
[root@linuxea.com /data/]*# cd harbor
[root@linuxea.com /data/harbor/]*# sed -i 's/ui_url_protocol = http/ui_url_protocol = https/'
harbor.cfg
[root@linuxea.com /data/harbor/]*# sed -i 's/hostname = reg.mydomain.com/hostname =
linuxea.reg.com/' harbor.cfg
[root@linuxea.com /data/harbor/]*# sed -i 's@ssl_cert = /data/cert/server.crt@ssl_cert =
/data/cert/linuxea.reg.com.crt@g' harbor.cfg
[root@linuxea.com /data/harbor/]*# sed -i 's@ssl_cert_key = /data/cert/server.key@ssl_cert_key =
/data/cert/linuxea.reg.com.key@g' harbor.cfg
[root@DS-VM-Node49 /data/harbor/]*# ./prepare
loaded secret key
Generated configuration file: ./common/config/nginx/nginx.conf
Generated configuration file: ./common/config/ui/env
Generated configuration file: ./common/config/ui/app.conf
Generated configuration file: ./common/config/registry/config.yml
Generated configuration file: ./common/config/db/env
Generated configuration file: ./common/config/jobservice/env
Generated configuration file: ./common/config/jobservice/app.conf
Generated configuration file: ./common/config/ui/private_key.pem
Generated configuration file: ./common/config/registry/root.crt
The configuration files are ready, please use docker-compose to start the service.
```

执行`.install` 会运行`docker-compose`，其中会去`dockerhub`拉取镜像，如果链接`dockerhub`较卡，可使用离线安装包

```
[root@linuxea.com /data/harbor]# ./install.sh
```

```
[Step 0]: checking installation environment ...
```

```
Note: docker version: 1.12.3
```

```
Note: docker-compose version: 1.8.0
```

```
[Step 1]: preparing environment ...
```

```
loaded secret key
```

```
Clearing the configuration file: ./common/config/jobservice/app.conf
```

```
Clearing the configuration file: ./common/config/jobservice/env
```

```
Clearing the configuration file: ./common/config/nginx/nginx.conf
```

```
Clearing the configuration file: ./common/config/nginx/cert/linuxea.reg.com.crt
```

```
Clearing the configuration file: ./common/config/nginx/cert/linuxea.reg.com.key
```

```
Clearing the configuration file: ./common/config/ui/app.conf
```

```
Clearing the configuration file: ./common/config/ui/private_key.pem
```

```
Clearing the configuration file: ./common/config/ui/env
```

```
Clearing the configuration file: ./common/config/db/env
```

```
Clearing the configuration file: ./common/config/registry/config.yml
```

```
Clearing the configuration file: ./common/config/registry/root.crt
```

```
Generated configuration file: ./common/config/nginx/nginx.conf
```

```
Generated configuration file: ./common/config/ui/env
```

```
Generated configuration file: ./common/config/ui/app.conf
```

```
Generated configuration file: ./common/config/registry/config.yml
```

```
Generated configuration file: ./common/config/db/env
```

```
Generated configuration file: ./common/config/jobservice/env
```

```
Generated configuration file: ./common/config/jobservice/app.conf
```

```
Generated configuration file: ./common/config/ui/private_key.pem
```

```
Generated configuration file: ./common/config/registry/root.crt
```

```
The configuration files are ready, please use docker-compose to start the service.
```

```
[Step 2]: checking existing instance of Harbor ...
```

```
[Step 3]: starting Harbor ...
```

```
Creating network "harbor_default" with the default driver
```

```
Pulling log (vmware/harbor-log:0.5.0)...
```

```
0.5.0: Pulling from vmware/harbor-log
```

```
93b3dcee11d6: Pull complete
```

```
5c14e4bdfb0b: Pull complete
```

```
e4dee8a574ca: Pull complete
```

```
64897311fadc: Pull complete
```

```
9b18f82d0181: Pull complete
```

```
Digest: sha256:81db268fa32ed35274f88f0e9f6e8c8fe36fdfe3247a7c75cef3d526814755ad
```

```
Status: Downloaded newer image for vmware/harbor-log:0.5.0
```

```
Pulling ui (vmware/harbor-ui:0.5.0)...
```

```
0.5.0: Pulling from vmware/harbor-ui
```

```
a91de92f52f5: Pull complete
```

```
341612b46e3b: Pull complete
```

c1a53f812656: Pull complete
9a9aa413559a: Pull complete
e1b377650dfc: Pull complete
be5517028022: Pull complete
726f4e0b4799: Pull complete
08639f4ec97a: Pull complete
Digest: sha256:677776af19c774f665565486ef0ec3ab37e55c6738a471942e3ea841b3e1821c
Status: Downloaded newer image for vmware/harbor-ui:0.5.0
Pulling mysql (vmware/harbor-db:0.5.0)...
0.5.0: Pulling from vmware/harbor-db
43c265008fae: Pull complete
d7abd54d3b34: Pull complete
92b527830a1b: Pull complete
44839710d611: Pull complete
3828a16bed5c: Pull complete
fb91763f6b4e: Pull complete
892bfb27c685: Pull complete
02874ec7a2dc: Pull complete
861c1296cc0d: Pull complete
d611998d5598: Pull complete
09037dc5a941: Pull complete
448973dd2180: Pull complete
c0a51ede01de: Pull complete
54c4b53ba168: Pull complete
Digest: sha256:4bc45566b8aab9288e76ac8a36e604aecf05ba9b25e22f5a9cd5e9686978b78b
Status: Downloaded newer image for vmware/harbor-db:0.5.0
Pulling jobservice (vmware/harbor-jobservice:0.5.0)...
0.5.0: Pulling from vmware/harbor-jobservice

93b3dcee11d6: Already exists
cdeb4de41efe: Pull complete
ce3594a70659: Pull complete
Digest: sha256:9c2d927f6e59fafcc930a5f738d713bde7d101cecd4e110a570a7b499d69ff68
Status: Downloaded newer image for vmware/harbor-jobservice:0.5.0
Pulling registry (library/registry:2.5.0)...
2.5.0: Pulling from library/registry

2ee5ed28ffa7: Pull complete
d1562c23a8aa: Pull complete
06ba8e23299f: Pull complete
802d2a9c64e8: Pull complete
Digest: sha256:1b68f0d54837c356e353efb04472bc0c9a60ae1c8178c9ce076b01d2930bcc5d
Status: Downloaded newer image for registry:2.5.0
Pulling proxy (nginx:1.11.5)...
1.11.5: Pulling from library/nginx
386a066cd84a: Pull complete
7bdb4b002d7f: Pull complete
49b006ddea70: Pull complete
Digest: sha256:9038d5645fa5fcca445d12e1b8979c87f46ca42cfb17beb1e5e093785991a639
Status: Downloaded newer image for nginx:1.11.5
Creating harbor-log
Creating harbor-ui
Creating registry

```
Creating harbor-db  
Creating harbor-jobservice  
Creating nginx
```

```
ERROR: for proxy Cannot create container for service proxy: Conflict. The name "/nginx" is  
already in use by container b88cbd48d16a3e52c77b24763bb241063ed2fb94eca467308f7fab31a1fbd403. You  
have to remove (or rename) that container to be able to reuse that name.
```

```
ERROR: Encountered errors while bringing up the project.
```

```
[root@linuxea.com /data/harbor]#
```

最后上面报错说容器已经存在，我们将以前的nginx容器删掉，然后重新./install

```
[root@linuxea.com /data/harbor]# docker rm nginx
nginx
[root@linuxea.com /data/harbor]# ./install.sh
```

[Step 0]: checking installation environment ...

Note: docker version: 1.12.3

Note: docker-compose version: 1.8.0

[Step 1]: preparing environment ...

loaded secret key

Clearing the configuration file: ./common/config/jobservice/app.conf

Clearing the configuration file: ./common/config/jobservice/env

Clearing the configuration file: ./common/config/nginx/nginx.conf

Clearing the configuration file: ./common/config/nginx/cert/linuxea.reg.com.crt

Clearing the configuration file: ./common/config/nginx/cert/linuxea.reg.com.key

Clearing the configuration file: ./common/config/ui/app.conf

Clearing the configuration file: ./common/config/ui/private_key.pem

Clearing the configuration file: ./common/config/ui/env

Clearing the configuration file: ./common/config/db/env

Clearing the configuration file: ./common/config/registry/config.yml

Clearing the configuration file: ./common/config/registry/root.crt

Generated configuration file: ./common/config/nginx/nginx.conf

Generated configuration file: ./common/config/ui/env

Generated configuration file: ./common/config/ui/app.conf

Generated configuration file: ./common/config/registry/config.yml

Generated configuration file: ./common/config/db/env

Generated configuration file: ./common/config/jobservice/env

Generated configuration file: ./common/config/jobservice/app.conf

Generated configuration file: ./common/config/ui/private_key.pem

Generated configuration file: ./common/config/registry/root.crt

The configuration files are ready, please use docker-compose to start the service.

[Step 2]: checking existing instance of Harbor ...

Note: stopping existing Harbor instance ...

Stopping harbor-jobservice ... done

Stopping harbor-db ... done

Stopping registry ... done

Stopping harbor-ui ... done

Stopping harbor-log ... done

Removing harbor-jobservice ... done

Removing harbor-db ... done

Removing registry ... done

Removing harbor-ui ... done

Removing harbor-log ... done

Removing network harbor_default

[Step 3]: starting Harbor ...

```
Creating network "harbor_default" with the default driver
Creating harbor-log
Creating harbor-db
Creating harbor-ui
Creating registry
Creating harbor-jobservice
Creating nginx

?.----Harbor has been installed and started successfully.----

Now you should be able to visit the admin portal at https://linuxea.reg.com.
For more details, please visit https://github.com/vmware/harbor .

[root@linuxea.com /data/harbor]#
```

3.1 web登陆和创建项目

admin / Harbor12345



摘要

我的项目:	1
我的镜像仓库:	0
公开项目:	1
公开镜像仓库:	0
全部项目:	1
全部镜像仓库:	0

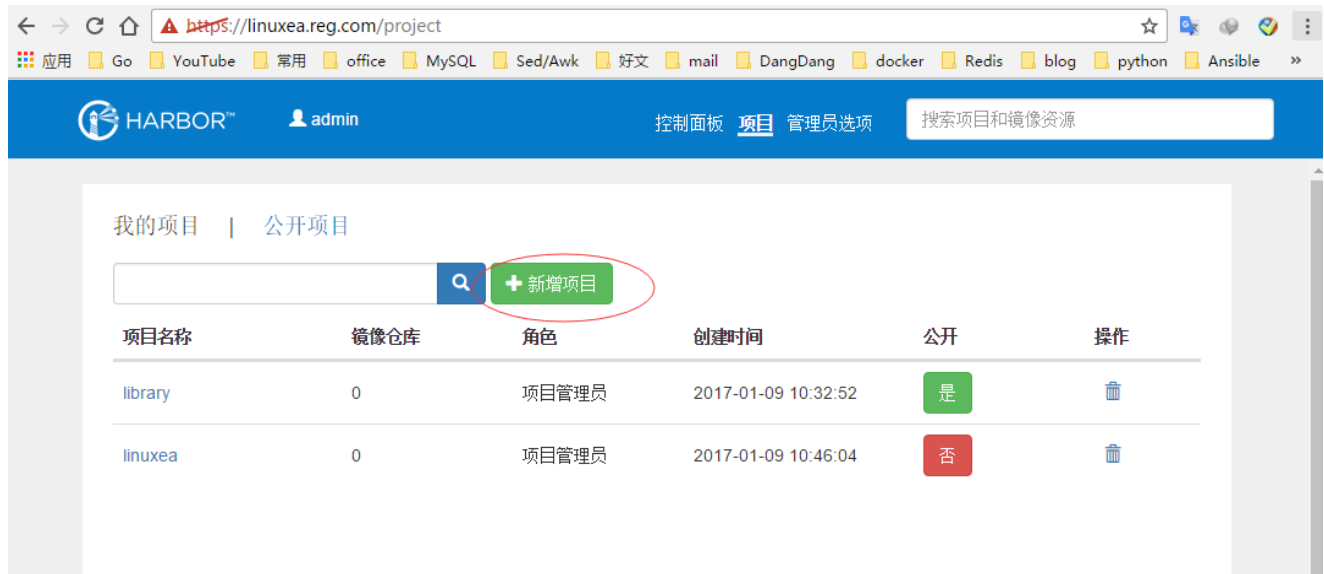
热门镜像仓库

镜像仓库名	下载次数
暂时没有数据。	

日志

用户名	镜像仓库名	标签	操作	时间戳
暂时没有数据。				

创建项目



我的项目 | 公开项目

搜索 + 新增项目

项目名称	镜像仓库	角色	创建时间	公开	操作
library	0	项目管理员	2017-01-09 10:32:52	是	
linuxea	0	项目管理员	2017-01-09 10:46:04	否	

4.push

尝试登陆:

```
[root@linuxea.com /data/harbor]# docker login -u admin -p Harbor12345 linuxea.reg.com
Error response from daemon: Get https://linuxea.reg.com/v1/users/: x509: certificate signed by
unknown authority
```

509: certificate signed by unknown authority 解决如下:

```
[root@linuxea.com /data/harbor]# mkdir -p /etc/docker/certs.d/linuxea.reg.com
[root@linuxea.com /data/harbor]# cp /data/cert/ca.crt /etc/docker/certs.d/linuxea.reg.com
[root@linuxea.com /data/harbor]# docker login -u admin -p Harbor12345 linuxea.reg.com
Login Succeeded
[root@linuxea.com /data/harbor]#
```

打包上传

```
[root@DS-VM-Node57 /data/harbor]# docker tag kep linuxea.reg.com/linuxea/keepalived
[root@DS-VM-Node57 /data/harbor]# docker push linuxea.reg.com/linuxea/keepalived
The push refers to a repository [linuxea.reg.com/linuxea/keepalived]
78c3ec04b8b3: Pushed
7cbcbac42c44: Pushed
latest: digest: sha256:f0382263e44b1d9fc7233f44b6cd92652f9fecf8e44dfd55ec49c05fed553ffb size: 739
[root@DS-VM-Node57 /data/harbor]# docker tag docker_nginx linuxea.reg.com/linuxea/nginx:1.11
[root@DS-VM-Node57 /data/harbor]# docker push linuxea.reg.com/linuxea/nginx
The push refers to a repository [linuxea.reg.com/linuxea/nginx]
ab0f3282dd72: Pushed
dbe4e1a4c160: Pushed
93d40cd73394: Pushed
4fe15f8d0ae6: Pushed
1.11: digest: sha256:6d7bc5796fe8d1259459762eb8423f34d1a6bf48b1617c6d6f0b58e4194e19df size: 1158
[root@DS-VM-Node57 /data/harbor]#
```

查看：

The screenshot shows the Harbor web interface in a browser. The address bar displays `https://linuxea.reg.com/repository#/repositories?project_id=2&is_public=0`. The interface includes a top navigation bar with the Harbor logo, user 'admin', and links for '控制面板', '项目', and '管理员选项'. A search bar is also present. The main content area shows the 'linuxea' project with a search bar and a list of repositories. Two repositories are listed: 'linuxea/keepalived' and 'linuxea/nginx'. The 'linuxea/nginx' repository is selected, showing its details. The table below lists the tags and the pull command for the '1.11' tag.

标签	镜像明细	Pull 命令	操作
latest	i	<code>docker pull linuxea.reg.com/linuxea/keepalived:latest</code>	🗑️

标签	镜像明细	Pull 命令	操作
1.11	i	<code>docker pull linuxea.reg.com/linuxea/nginx:1.11</code>	🗑️

5.pull

换一台机器pull

1, 首先需要harbor的ca信任, 创建路径后, 在harbor将ca传递过来

```
[root@LinuxEA-COM /data/harbor/cert]# mkdir -p /etc/docker/certs.d/linuxea.reg.com
```

2, 在harbor上把ca复制到pull的机器上

```
[root@LinuxEA-COM /data/harbor]# scp /data/cert/ca.crt
10.0.1.49:/etc/docker/certs.d/linuxea.reg.com
The authenticity of host '10.0.1.49 (10.0.1.49)' can't be established.
ECDSA key fingerprint is 0b:26:0a:e4:1f:04:b5:71:64:a3:0f:e6:e7:11:ee:b0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.1.49' (ECDSA) to the list of known hosts.
root@10.0.1.49's password:
ca.crt
100% 1935    1.9KB/s   00:00
```

3, 非公开项目需要登陆, 公开项目不需要登陆

```
[root@LinuxEA-COM /data/harbor/cert]# docker login -u admin -p Harbor12345 linuxea.reg.com
WARNING: Error loading config file:/root/.docker/config.json - EOF
Login Succeeded
```

4, 开始pull

```
[root@LinuxEA-COM /data/harbor/cert]# docker pull linuxea.reg.com/linuxea/nginx:1.11
1.11: Pulling from linuxea/nginx

e110a4a17941: Already exists
a91b6045bb20: Already exists
8e1413698bdb: Pull complete
Digest: sha256:6d7bc5796fe8d1259459762eb8423f34d1a6bf48b1617c6d6f0b58e4194e19df
Status: Downloaded newer image for linuxea.reg.com/linuxea/nginx:1.11
```

5, 查看

```
[root@LinuxEA-COM /data/harbor/cert]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
linuxea.reg.com/linuxea/nginx	1.11	7543cf287f61	9 days ago	231.2 MB
10.0.1.49/keepalived	latest	7bf39df0e969	5 days ago	11.73 MB
keepalived_keepalived	latest	7bf39df0e969	5 days ago	11.73 MB
haproxy_haproxy	latest	a3e7eec9950f	6 days ago	282.8 MB
10.0.1.49:5000/nginx_83f8645	latest	7c2b1677f82e	8 days ago	225.5 MB
10.0.1.49/nginx_83f8645	latest	7c2b1677f82e	8 days ago	225.5 MB
10.0.1.49:5000/nginx_3d56146	latest	7e8691bb5e6c	8 days ago	225.5 MB
10.0.1.49:5000/nginx_c98a1b2	latest	659dfd0f871c	8 days ago	225.5 MB
10.0.1.49:5000/nginx_03fef24	latest	0a289aac1bdc	9 days ago	225.5 MB
10.0.1.49:5000/nginx_b2cbcd8	latest	f0a1b955313c	9 days ago	225.5 MB
10.0.1.49:5000/nginx_eb5b609	latest	e4f7adaf8eab	9 days ago	225.5 MB
10.0.1.49:5000/nginx_40629bd	latest	3408138f55e5	9 days ago	225.5 MB
10.0.1.49:5000/nginx_9c104e3	latest	743caf8e9447	9 days ago	225.5 MB
registry	latest	182810e6ba8c	12 days ago	37.62 MB
docker_svb1.9.4	latest	331d234e3bbd	13 days ago	14.46 MB
tomcat_tomcat	latest	eda0380486d8	2 weeks ago	728.7 MB
jenkins	latest	f61635651954	2 weeks ago	714.2 MB
docker_php-fpm	latest	ded2b5ca5aaa	3 weeks ago	796 MB

```
[root@LinuxEA-COM /data/harbor/cert]#
```

出现 `unauthorized: authentication required` 解决办法，一般在web中创建了项目是不会出现的

则可以在 `/data/harborcommon/config/nginx/nginx.conf` 中添加 `proxy_set_header X-Forwarded-Proto https;`