参照: <https://github.com/dcm4che/dcm4chee-arc-light/wiki/Run-secured-archive-services-and-Elastic-Stack-on-a-single-host>

虚拟机(Docker Host)环境：需要有2G内存，以及15G以上的磁盘空间，本人试过内存512M有些容器是起不来的。

1. 下载与安装Docker：
2. yum remove docker docker-common docker-selinux docker-engine
3. yum install -y yum-utils device-mapper-persistent-data lvm2
4. yum-config-manager -y --add-repo <https://download.docker.com/linux/centos/docker-ce.repo>
5. yum install -y docker-ce
6. systemctl start docker
7. systemctl enable docker
8. 下载与安装docker-compose：

curl -L "https://github.com/docker/compose/releases/download/1.24.1/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

chmod +x /usr/local/bin/docker-compose

1. 新建一个目录，如/pacs，并进入目录。
2. touch docker-compose.yml docker-compose.env
3. 把<https://github.com/dcm4che/dcm4chee-arc-light/wiki/Run-secured-archive-services-and-Elastic-Stack-on-a-single-host>上面的内容分别对应复制到这两个文件。

可能需要修改里面hostname，ip之类的。

1. mkdir -p /var/local/dcm4chee-arc/logstash

chown 1000:1000 /var/local/dcm4chee-arc/logstash

1. echo "Asia/shanghai" > /etc/timezone

cp /usr/share/zoneinfo/Asia/Shanghai /etc/localtime

1. docker-compose -p dcm4chee up –d

compose完之后看到以下镜像和容器：

[root@pad ~]# docker image ls

REPOSITORY TAG IMAGE ID CREATED SIZE

dcm4che/dcm4chee-arc-psql 5.18.0-secure 35c287cc5d74 10 days ago 585MB

dcm4che/slapd-dcm4chee 2.4.44-18.0 732f81fb3645 10 days ago 164MB

dcm4che/keycloak 6.0.1 7c916907b9c2 10 days ago 638MB

dcm4che/logstash-dcm4chee 7.2.0-9 0c8c6197ed63 12 days ago 880MB

docker.elastic.co/kibana/kibana-oss 7.2.0 1e03c8c0c06e 7 weeks ago 594MB

docker.elastic.co/elasticsearch/elasticsearch-oss 7.2.0 1d6c081f3e56 7 weeks ago 768MB

dcm4che/keycloak-gatekeeper 6.0.1 a560f064090a 3 months ago 19.2MB

dcm4che/postgres-dcm4chee 11.2-16 c5949c05c8b6 5 months ago 312MB

[root@pad pacs]# docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

90f0fb6a41a5 dcm4che/dcm4chee-arc-psql:5.18.0-secure "/docker-entrypoint.…" 2 minutes ago Up About a minute 0.0.0.0:2575->2575/tcp, 0.0.0.0:8080->8080/tcp, 0.0.0.0:8443->8443/tcp, 0.0.0.0:9990->9990/tcp, 0.0.0.0:11112->11112/tcp dcm4chee\_arc\_1

4759147f0386 dcm4che/keycloak-gatekeeper:6.0.1 "/opt/keycloak-gatek…" 2 minutes ago Up About a minute 0.0.0.0:8643->8643/tcp dcm4chee\_keycloak-gatekeeper\_1

d55535a880ec dcm4che/postgres-dcm4chee:11.2-16 "docker-entrypoint.s…" 2 minutes ago Up 2 minutes 0.0.0.0:5432->5432/tcp dcm4chee\_db\_1

4895bdded24d dcm4che/keycloak:6.0.1 "/docker-entrypoint.…" 2 minutes ago Up 2 minutes 0.0.0.0:8843->8843/tcp, 0.0.0.0:8880->8880/tcp, 0.0.0.0:8990->8990/tcp dcm4chee\_keycloak\_1

a4d228d4b9c8 dcm4che/logstash-dcm4chee:7.2.0-9 "/docker-entrypoint.…" 2 minutes ago Up 2 minutes 5044/tcp, 0.0.0.0:8514->8514/udp, 9600/tcp, 0.0.0.0:8514->8514/tcp, 0.0.0.0:12201->12201/udp dcm4chee\_logstash\_1

6694236f8e44 docker.elastic.co/kibana/kibana-oss:7.2.0 "/usr/local/bin/kiba…" 2 minutes ago Up 2 minutes 5601/tcp dcm4chee\_kibana\_1

e89cd091419f dcm4che/slapd-dcm4chee:2.4.44-18.0 "/docker-entrypoint.…" 2 minutes ago Up 2 minutes 0.0.0.0:389->389/tcp dcm4chee\_ldap\_1

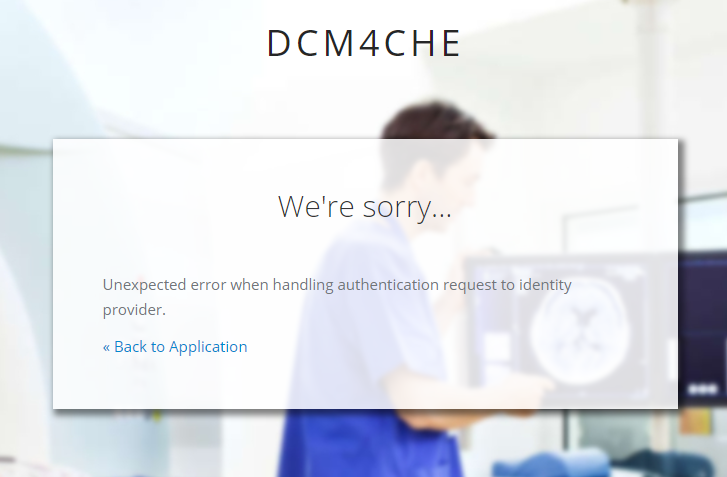
c5438617167b docker.elastic.co/elasticsearch/elasticsearch-oss:7.2.0 "/usr/local/bin/dock…" 2 minutes ago Up 7 seconds 0.0.0.0:9200->9200/tcp, 0.0.0.0:9300->9300/tcp dcm4chee\_elasticsearch\_1

1. 把Archive UI注册到Keycloak的OpenID Connect Client：
2. 浏览器打开：

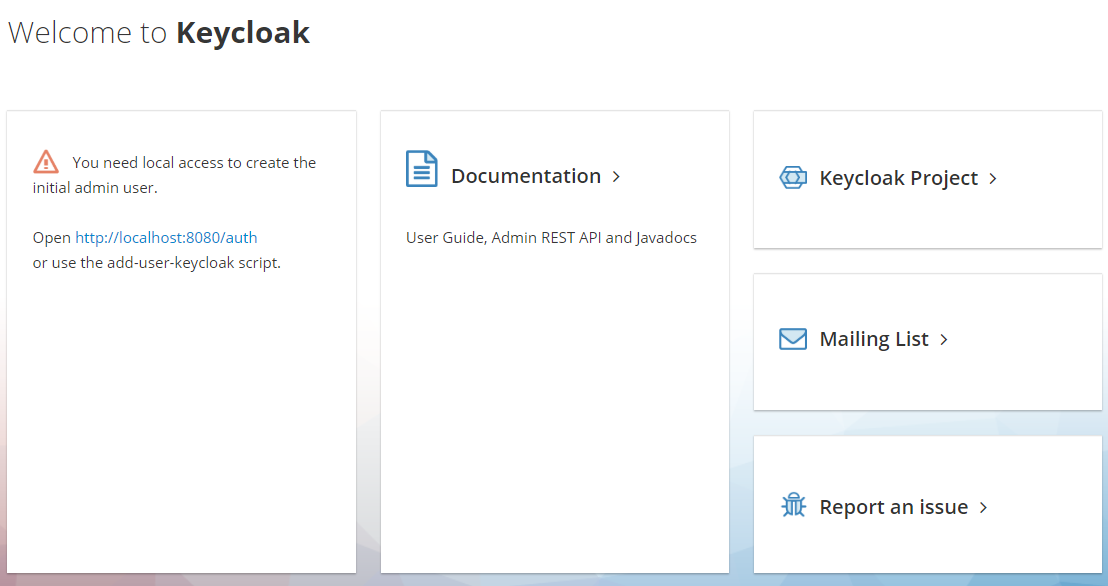
https://172.16.0.19:8843/auth/admin/dcm4che/console



然而输入用户名：admin，密码：admin出现了如下错误信息：



<https://172.16.0.19:8843/auth/>



另外，本人把镜像删除重新拉取后，发现无法访问页面，无论访问什么都是refuse。

有像：

Aug 13 09:16:51 pad dockerd: time="2019-08-13T09:16:51.328806915+08:00" level=error msg="Failed to log msg \"\" for logger gelf: gelf: cannot send GELF message: w

rite udp 127.0.0.1:44350->127.0.0.1:12201: write: connection refused"

这样的信息。

而且运行当中有一些容器又会自动退出。

同时docker-compose up之后

[root@pad pacs]# docker-compose ps

Name Command State Ports

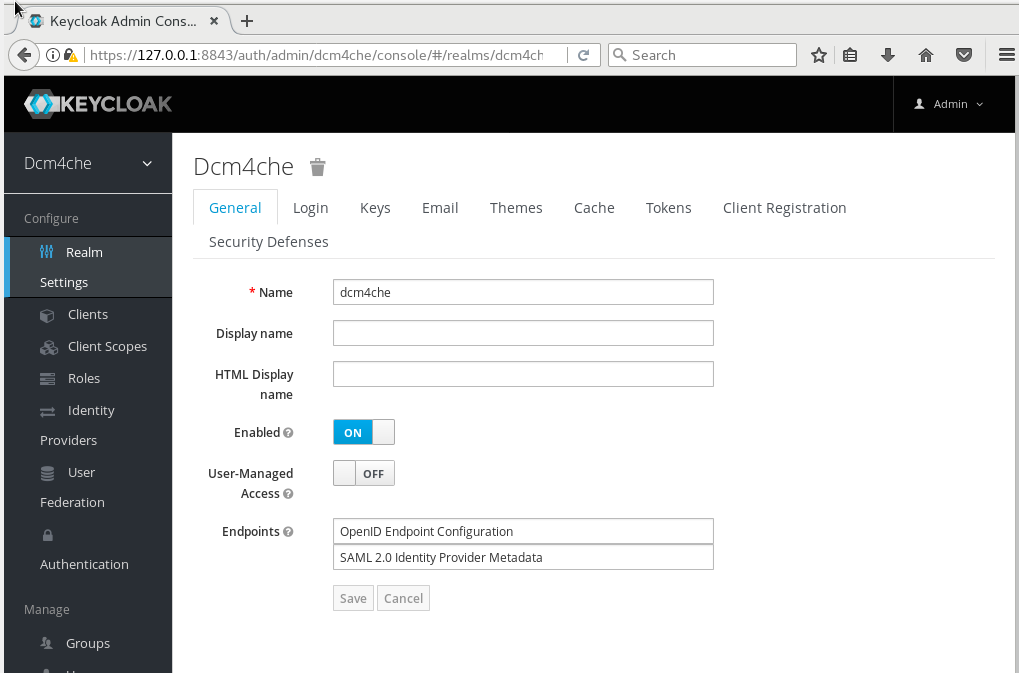
------------------------------

看不到输出。

在docker-compose up的时候看到以下信息：

Aug 15 10:42:03 pad dockerd: time="2019-08-15T10:42:03.127620345+08:00" level=info msg="No non-localhost DNS nameservers are left in resolv.conf. Using default external servers: [nameserver 8.8.8.8 nameserver 8.8.4.4]"

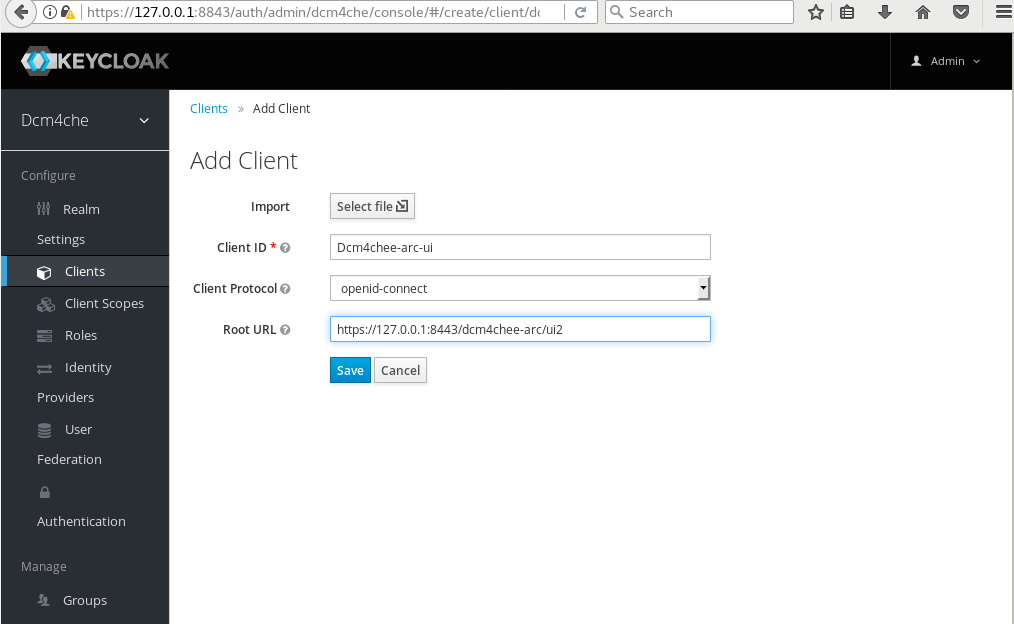
1. 重新安装Linux系统，该Linux系统需要自带图形界面。重新按照上面步骤，发现可以登录：



把Archive UI注册到Keycloak的OpenID Connect Client：

<https://127.0.0.1:8843/auth/admin/dcm4che/console>

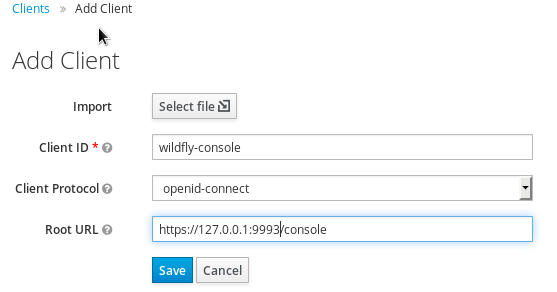
添加Client：



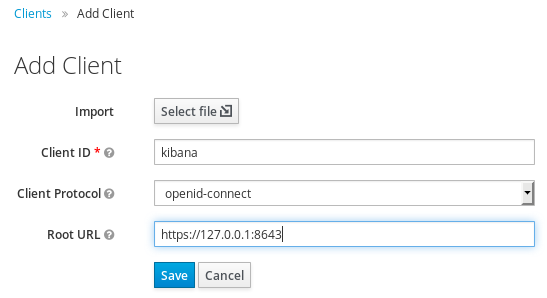
需要添加一些URL：

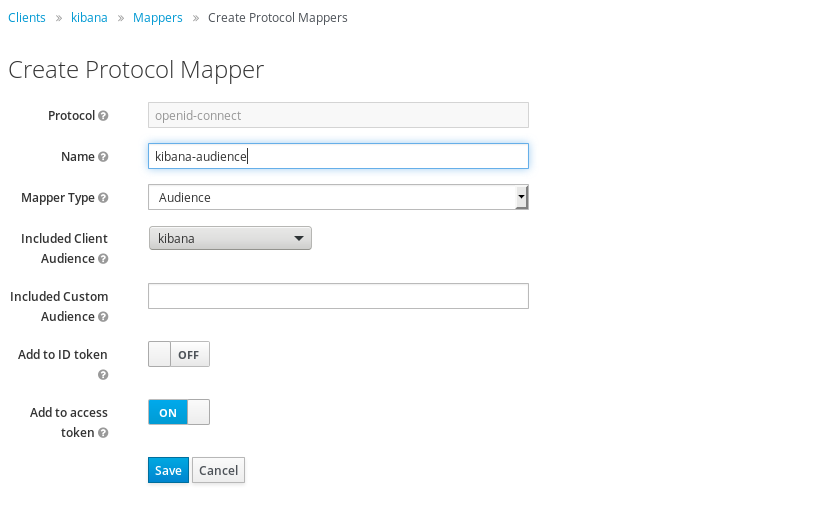


1. Register the WildFly Administration Console as OIDC client in Keycloak：

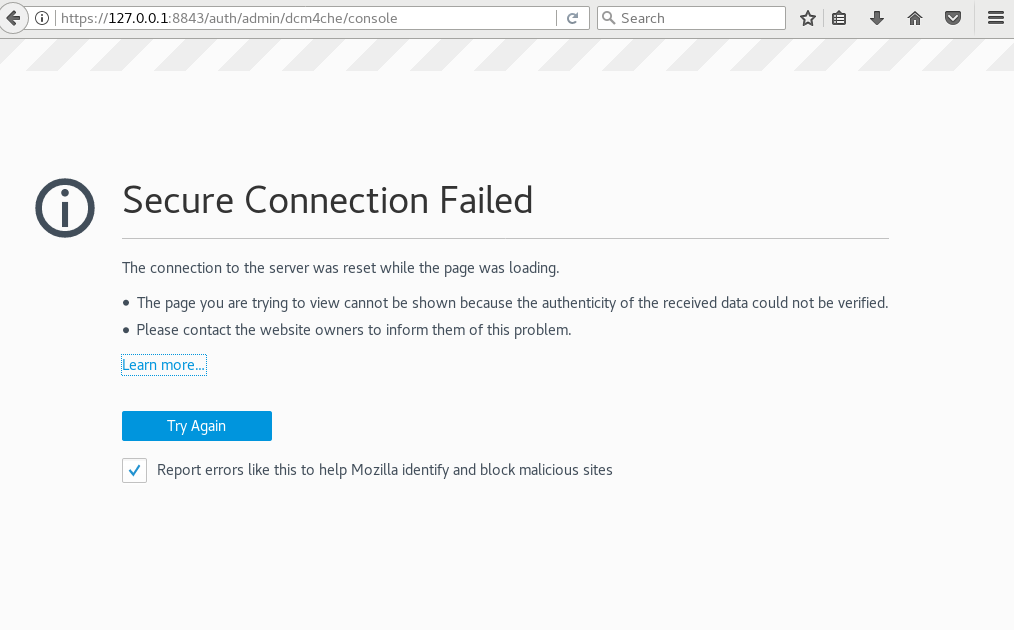


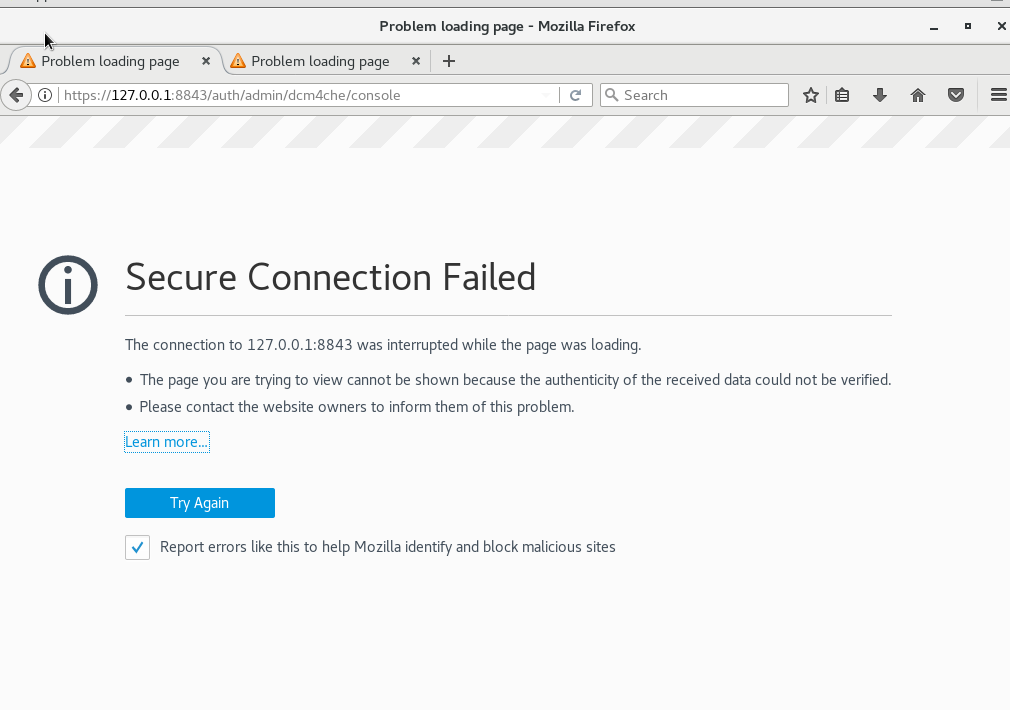
1. Register Keyclock Gatekeeper as OIDC client in Keycloak

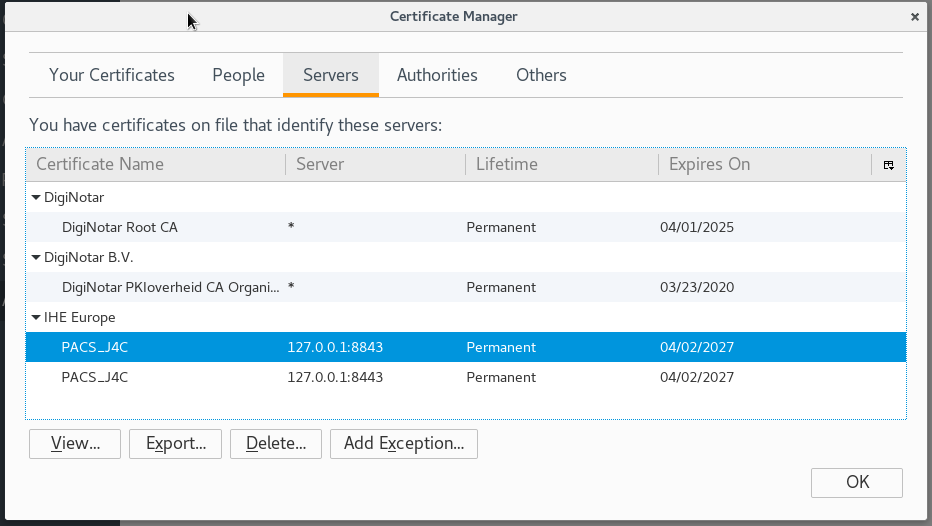




1. 然而，停了docker-compose重新拉起又无法访问：







弄了也没有效果的。

以及在原来pad（一开始没有配置图形界面）。在docker映射的持久化存储/var/local/dcm4chee-arc/把一些配置文件中原来的127.0.0.1都改成0.0.0.0都没有效果。还是refuse。如果到容器里面的serverlog查看，并没有看到拒绝的信息。

注意到的奇怪现象：1.第一次使用docker-compose都能访问到页面，之后把容器down了之后重新拉起就无法访问，清理docker缓存，删除持久化存储都无法；2.即便后来在持久化存储把配置文件的127.0.0.1都改成0.0.0.0还是无法访问，而且容器里面日志都没有拒绝连接的相关输出；3.刚开始八个容器，然后逐渐会有一些容器自动退出。

现在把容器逐个按顺序启动，进入容器观察：

1. [root@localhost pacs]# docker network create dcm4chee\_default

426b4c4a1a363eb6d70ff04cc0d549b9f13c06c0ccb83e24872948b2b00dbecb

2.

docker run --network=dcm4chee\_default --name elasticsearch \

-e ES\_JAVA\_OPTS="-Xms1024m -Xmx1024m" \

-e TAKE\_FILE\_OWNERSHIP=1 \

-e discovery.type=single-node \

-p 9200:9200 \

-p 9300:9300 \

-v /etc/localtime:/etc/localtime:ro \

-v /etc/timezone:/etc/timezone:ro \

-v /var/local/dcm4chee-arc/elasticsearch:/usr/share/elasticsearch/data \

-d docker.elastic.co/elasticsearch/elasticsearch-oss:7.2.0

[root@localhost pacs]# docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

045a64248f1b docker.elastic.co/elasticsearch/elasticsearch-oss:7.2.0 "/usr/local/bin/dock…" 5 seconds ago Up 2 seconds 0.0.0.0:9200->9200/tcp, 0.0.0.0:9300->9300/tcp elasticsearch

[root@localhost ~]# docker port 045a64248f

9200/tcp -> 0.0.0.0:9200

9300/tcp -> 0.0.0.0:9300

[root@045a64248f1b elasticsearch]# netstat -naput

Active Internet connections (servers and established)

Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name

tcp 0 0 0.0.0.0:9200 0.0.0.0:\* LISTEN -

tcp 0 0 0.0.0.0:9300 0.0.0.0:\* LISTEN -

[root@045a64248f1b elasticsearch]# route

Kernel IP routing table

Destination Gateway Genmask Flags Metric Ref Use Iface

default gateway 0.0.0.0 UG 0 0 0 eth0

172.18.0.0 0.0.0.0 255.255.0.0 U 0 0 0 eth0

3.

docker run --network=dcm4chee\_default --name kibana \

-p 5601:5601 \

-v /etc/localtime:/etc/localtime:ro \

-v /etc/timezone:/etc/timezone:ro \

-d docker.elastic.co/kibana/kibana-oss:7.2.0

[root@localhost ~]# docker exec -it -u root 2534d7f836 /bin/bash

4.

docker run --network=dcm4chee\_default --name logstash \

-p 12201:12201/udp \

-p 8514:8514/udp \

-p 8514:8514 \

-v /etc/localtime:/etc/localtime:ro \

-v /etc/timezone:/etc/timezone:ro \

-v /var/local/dcm4chee-arc/logstash/filter-hashtree:/usr/share/logstash/data/filter-hashtree \

-d dcm4che/logstash-dcm4chee:7.2.0-9

tcp 0 0 0.0.0.0:8514 0.0.0.0:\* LISTEN -

udp 0 0 0.0.0.0:8514 0.0.0.0:\* -

udp 0 0 0.0.0.0:12201 0.0.0.0:\* -

5.

docker run --network=dcm4chee\_default --name ldap \

--log-driver gelf \

--log-opt gelf-address=udp://172.17.0.1:12201 \

--log-opt tag=slapd \

-p 389:389 \

-e SYSLOG\_HOST=logstash \

-e SYSLOG\_PORT=8514 \

-e SYSLOG\_PROTOCOL=TLS \

-v /etc/localtime:/etc/localtime:ro \

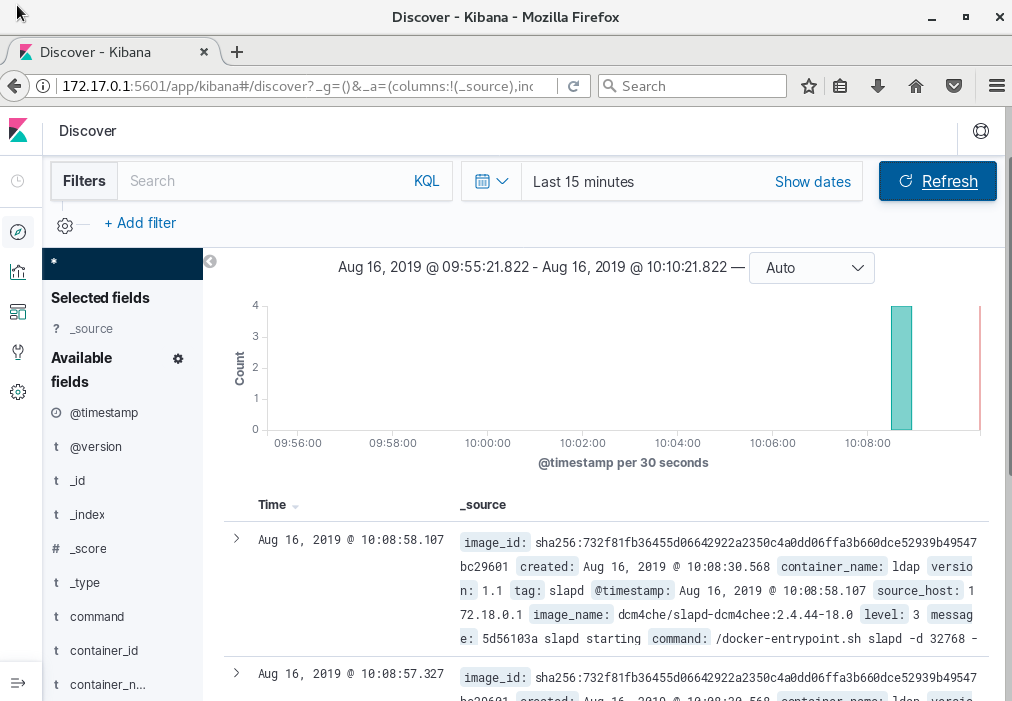
-v /etc/timezone:/etc/timezone:ro \

-v /var/local/dcm4chee-arc/ldap:/var/lib/ldap \

-v /var/local/dcm4chee-arc/slapd.d:/etc/ldap/slapd.d \

-d dcm4che/slapd-dcm4chee:2.4.44-18.0

在Kibana页面已经有显示容器相关日志信息：



主要是在左边有Selected fields（已选上的项目，会在右边），Available fields(可以把它添加到Selected fields)。以及可以对日志进行筛选。

注：Kibana的页面不需要在Docker host上才能浏览，在Windows上也能浏览。

6.

docker run --network=dcm4chee\_default --name keycloak \

--log-driver gelf \

--log-opt gelf-address=udp://172.17.0.1:12201 \

--log-opt tag=keycloak \

-p 8880:8880 \

-p 8843:8843 \

-p 8990:8990 \

-p 8993:8993 \

-e HTTP\_PORT=8880 \

-e HTTPS\_PORT=8843 \

-e MANAGEMENT\_HTTP\_PORT=8990 \

-e MANAGEMENT\_HTTPS\_PORT=8993 \

-e LOGSTASH\_HOST=logstash \

-e KEYCLOAK\_WAIT\_FOR="ldap:389 logstash:8514" \

-v /etc/localtime:/etc/localtime:ro \

-v /etc/timezone:/etc/timezone:ro \

-v /var/local/dcm4chee-arc/keycloak:/opt/keycloak/standalone \

-d dcm4che/keycloak:6.0.1

docker exec -it -u root 7e28efa3c91 /bin/bash

root@7e28efa3c91d:/# netstat -naput

Active Internet connections (servers and established)

Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name

tcp 0 0 127.0.0.11:39324 0.0.0.0:\* LISTEN -

tcp 0 0 172.18.0.6:8990 0.0.0.0:\* LISTEN -

tcp 0 0 172.18.0.6:8993 0.0.0.0:\* LISTEN -

tcp 0 0 0.0.0.0:42310 0.0.0.0:\* LISTEN -

tcp 0 0 172.18.0.6:8843 0.0.0.0:\* LISTEN -

tcp 0 0 172.18.0.6:8880 0.0.0.0:\* LISTEN -

tcp 0 0 172.18.0.6:57270 151.101.228.204:80 TIME\_WAIT -

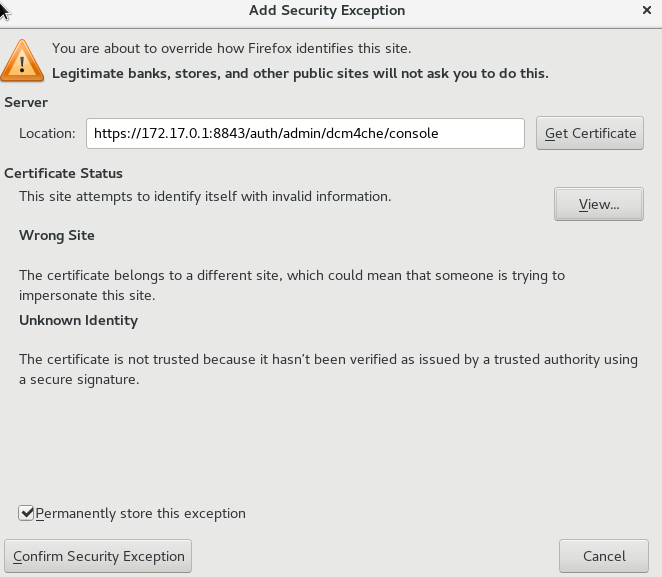
tcp 0 0 172.18.0.6:57138 151.101.228.204:80 TIME\_WAIT -

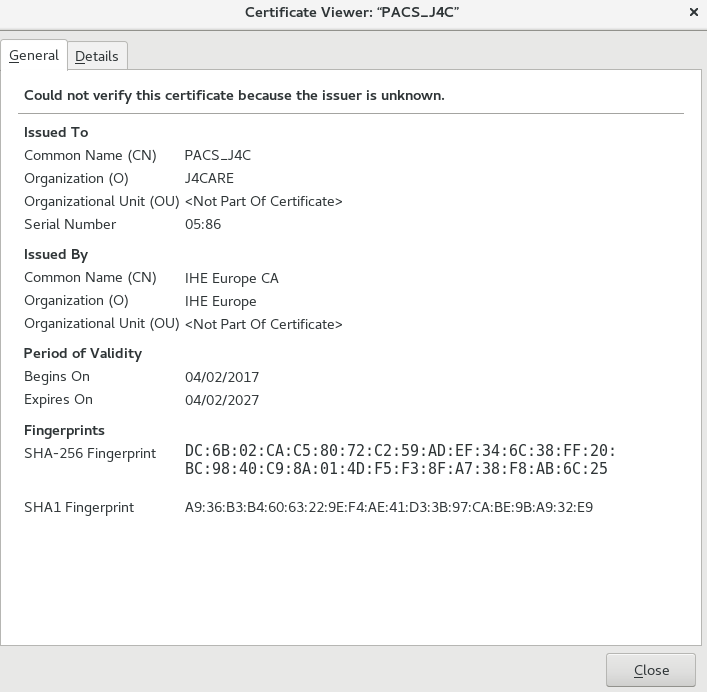
tcp 0 0 172.18.0.6:57140 151.101.228.204:80 TIME\_WAIT -

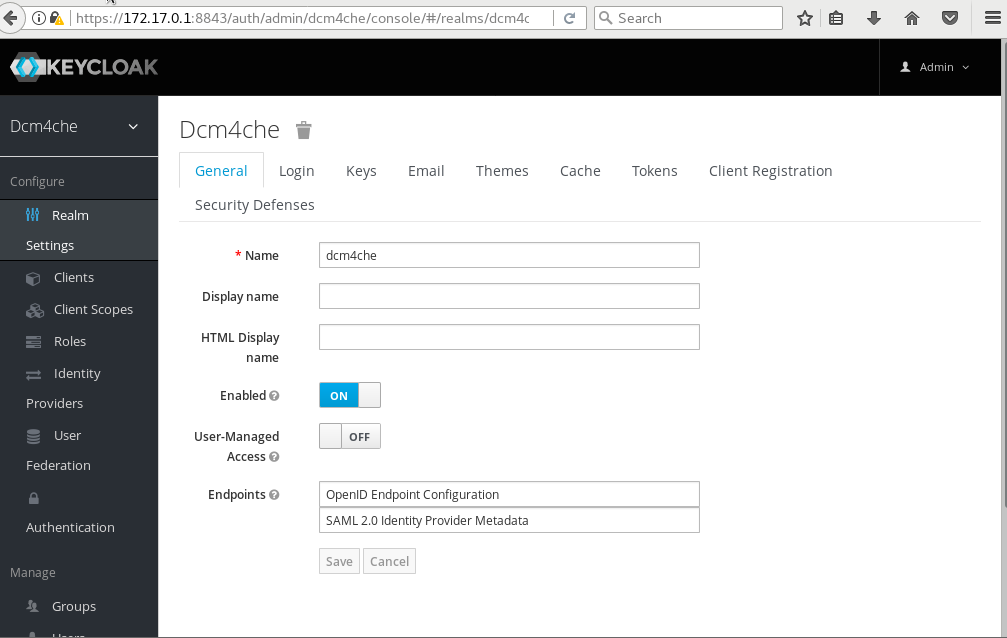
udp 0 0 127.0.0.11:46641 0.0.0.0:\* -

udp 0 0 172.18.0.6:55478 172.18.0.4:12201 ESTABLISHED -

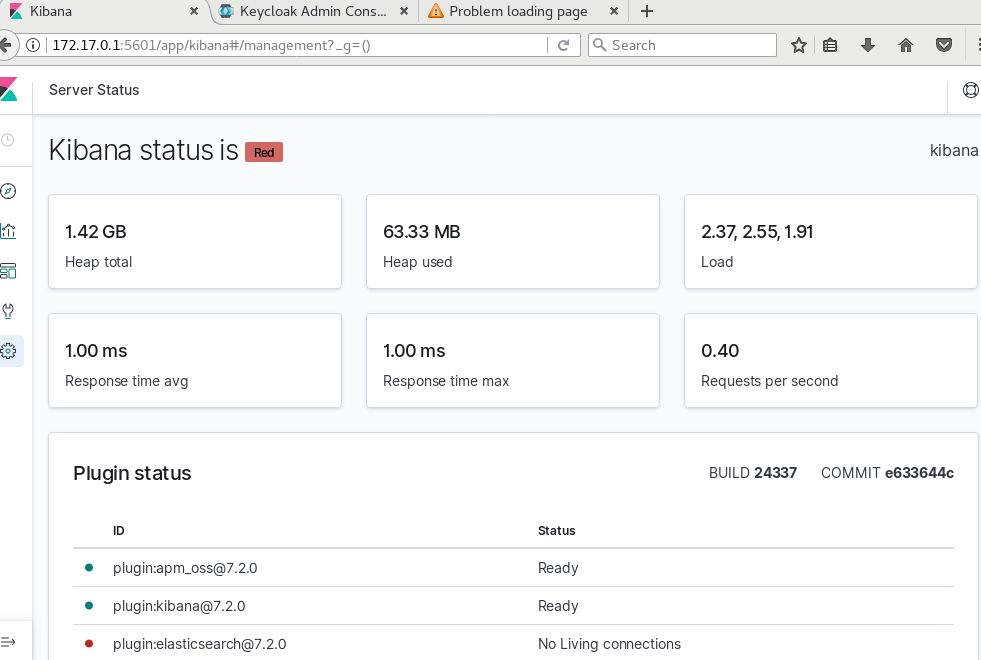
尝试登录：172.17.0.1:8843/auth/admin/dcm4che/console







然而此时Kibana出了问题，点什么都没反应：



7.

docker run --network=dcm4chee\_default --name db \

--log-driver gelf \

--log-opt gelf-address=udp://172.17.0.1:12201 \

--log-opt tag=postgres \

-p 5432:5432 \

-e POSTGRES\_DB=pacsdb \

-e POSTGRES\_USER=pacs \

-e POSTGRES\_PASSWORD=pacs \

-v /etc/localtime:/etc/localtime:ro \

-v /etc/timezone:/etc/timezone:ro \

-v /var/local/dcm4chee-arc/db:/var/lib/postgresql/data \

-d dcm4che/postgres-dcm4chee:11.2-16

docker exec -it -u root 3dcf0066be60f4 /bin/bash

8.

docker run --network=dcm4chee\_default --name arc \

--log-driver gelf \

--log-opt gelf-address=udp://172.17.0.1:12201 \

--log-opt tag=dcm4chee-arc \

-p 8080:8080 \

-p 8443:8443 \

-p 9990:9990 \

-p 9993:9993 \

-p 11112:11112 \

-p 2575:2575 \

-e POSTGRES\_DB=pacsdb \

-e POSTGRES\_USER=pacs \

-e POSTGRES\_PASSWORD=pacs \

-e LOGSTASH\_HOST=logstash \

-e WILDFLY\_WAIT\_FOR="ldap:389 db:5432 logstash:8514" \

-e AUTH\_SERVER\_URL=https://172.17.0.1:8843/auth \

-v /etc/localtime:/etc/localtime:ro \

-v /etc/timezone:/etc/timezone:ro \

-v /var/local/dcm4chee-arc/wildfly:/opt/wildfly/standalone \

-d dcm4che/dcm4chee-arc-psql:5.18.0-secure

然而可以看到elasticsearch已经自动退出了，

045a64248f1b docker.elastic.co/elasticsearch/elasticsearch-oss:7.2.0 "/usr/local/bin/dock…" 2 hours ago Exited (137) 20 minutes ago elasticsearch

试图重新拉起：

docker start 045a64

然而又自动退出：

045a64248f1b docker.elastic.co/elasticsearch/elasticsearch-oss:7.2.0 "/usr/local/bin/dock…" 2 hours ago Exited (1) 55 seconds ago elasticsearch

同时另一个容器也自动退出：

b765a8e03547 dcm4che/dcm4chee-arc-psql:5.18.0-secure "/docker-entrypoint.…" 7 minutes ago Exited (1) 36 seconds ago

本人估计是Docker Host或者是主机配置问题，导致容器不稳定，所以自动退出，有一个现象是在启动最后一个容器后虚拟机就变得很卡。

但后来把dcm4che/dcm4chee-arc-psql:5.18.0-secure还是可以拉起来的。

9.

docker run --network=dcm4chee\_default --name keycloak-gatekeeper \

--log-driver gelf \

--log-opt gelf-address=udp://172.17.0.1:12201 \

--log-opt tag=keycloak-gatekeaper \

-p 8643:8643 \

-e PROXY\_LISTEN=:8643 \

-e PROXY\_REDIRECTION\_URL=https://172.17.0.1:8643 \

-e PROXY\_UPSTREAM\_URL=http://kibana:5601 \

-e PROXY\_DISCOVERY\_URL=https://172.17.0.1:8843/auth/realms/dcm4che \

-e PROXY\_CLIENT\_ID=kibana \

-e PROXY\_CLIENT\_SECRET=968956a3-582b-4f8c-9d09-249fc9f74c35 \

-e PROXY\_ENCRYPTION\_KEY=AgXa7xRcoClDEU0ZDSH4X0XhL5Qy2Z2j \

-d dcm4che/keycloak-gatekeeper:6.0.1 \

--openid-provider-timeout=120s \

--tls-cert=/etc/certs/cert.pem \

--tls-private-key=/etc/certs/key.pem \

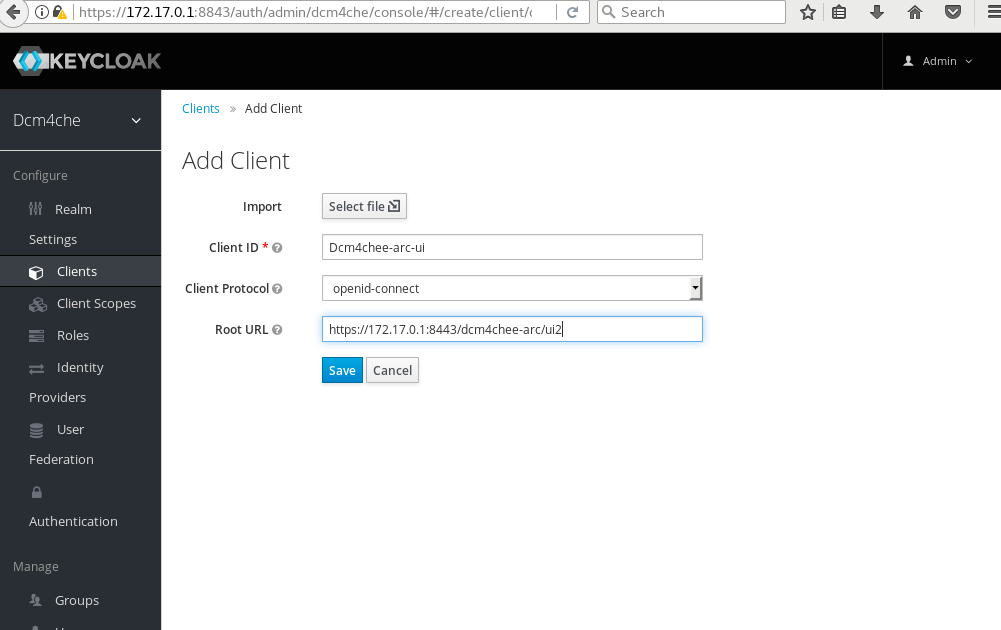
--skip-openid-provider-tls-verify=true \

--enable-refresh-tokens=true \

--resources=uri=/\*|roles=auditlog

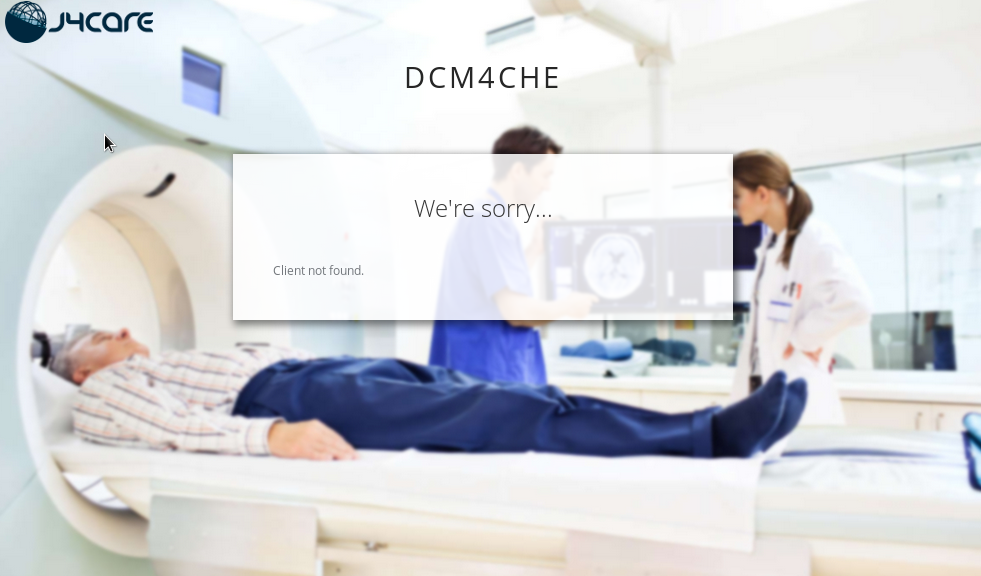
此时不知为何不卡。

10.





尝试登录dcm4chee报错：



解决办法：

1. 进入dcm4che/dcm4chee-arc-psql:5.18.0-secure容器。
2. 打开jboss.cli

[standalone@localhost:9993 /] /system-property=auth-server-url:remove

{"outcome" => "success"}

[standalone@localhost:9993 /] /system-property=auth-server-url:add(value=http://172.17.0.1:8080/auth)

{"outcome" => "success"}

1. cat /opt/wildfly/standalone/log/server.log |grep deploy

发现是：dcm4chee-arc-ear-5.18.0-psql-secure.ear

1. 再进入jboss.cli

[standalone@localhost:9993 /] /subsystem=keycloak/secure-deployment=dcm4chee-arc-ui2-5.18.0-secure.war/:remove

{"outcome" => "success"}

[standalone@localhost:9993 /] /subsystem=keycloak/secure-deployment=dcm4chee-arc-ui2-5.18.0-secure.war/:add(realm=dcm4che,resource=dcm4chee-arc-ui,auth-server-url=http://172.17.0.1:8080/auth,ssl-required=external,public-client=true)

{

"outcome" => "failed",

"failure-description" => "WFLYCTL0436: Cannot register capability 'org.wildfly.security.http-server-mechanism-factory.dcm4chee-arc-ui2-5.18.0-secure.war' at l

ocation '[

(\"subsystem\" => \"keycloak\"),

(\"secure-deployment\" => \"dcm4chee-arc-ui2-5.18.0-secure.war\")

]' as it is already registered in context 'global' at location(s) '[[

(\"subsystem\" => \"keycloak\"),

(\"secure-deployment\" => \"dcm4chee-arc-ui2-5.18.0-secure.war\")

]]'",

"rolled-back" => true

}

[standalone@localhost:9993 /] /subsystem=keycloak/secure-deployment=dcm4chee-arc-war-5.18.0-secure.war/:remove

{"outcome" => "success"}

[standalone@localhost:9993 /] /subsystem=keycloak/secure-deployment=dcm4chee-arc-war-5.18.0-secure.war/:add(realm=dcm4che,resource=dcm4chee-arc-rs,auth-server-url=http://172.17.0.1:8080/auth,ssl-required=external,bearer-only=true)

{

"outcome" => "failed",

"failure-description" => "WFLYCTL0436: Cannot register capability 'org.wildfly.security.http-server-mechanism-factory.dcm4chee-arc-war-5.18.0-secure.war' at l

ocation '[

(\"subsystem\" => \"keycloak\"),

(\"secure-deployment\" => \"dcm4chee-arc-war-5.18.0-secure.war\")

]' as it is already registered in context 'global' at location(s) '[[

(\"subsystem\" => \"keycloak\"),

(\"secure-deployment\" => \"dcm4chee-arc-war-5.18.0-secure.war\")

]]'",

"rolled-back" => true

}

先重启wildfly：

root@b765a8e03547:/opt/wildfly/bin# ./jboss-cli.sh --connect command=:reload

再：

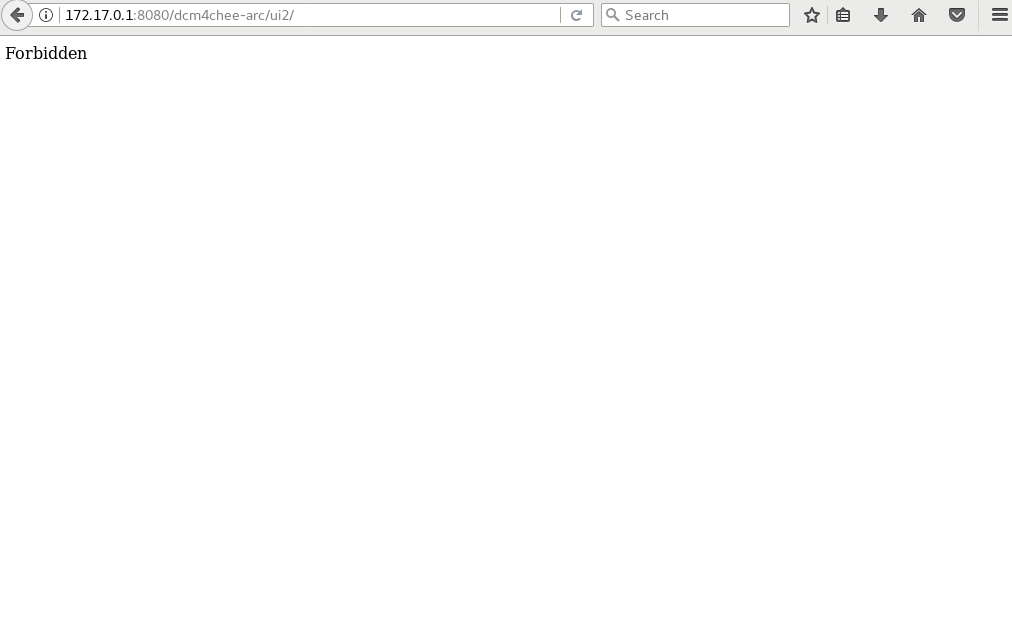
[standalone@localhost:9993 /] /subsystem=keycloak/secure-deployment=dcm4chee-arc-ui2-5.18.0-secure.war/:add(realm=dcm4che,resource=dcm4chee-arc-ui,auth-server-url=http://172.17.0.1:8080/auth,ssl-required=external,public-client=true)

{"outcome" => "success"}

[standalone@localhost:9993 /] /subsystem=keycloak/secure-deployment=dcm4chee-arc-war-5.18.0-secure.war/:add(realm=dcm4che,resource=dcm4chee-arc-rs,auth-server-url=http://172.17.0.1:8080/auth,ssl-required=external,bearer-only=true)

{"outcome" => "success"}

不过失败了：



试图重启keycloak又报错：

root@7e28efa3c91d:/opt/keycloak/bin# ./jboss-cli.sh --connect command=:reload

Failed to connect to the controller: The controller is not available at localhost:9990: java.net.ConnectException: WFLYPRT0053: Could not connect to remote+http://localhost:9990. The connection failed: WFLYPRT0053: Could not connect to remote+http://localhost:9990. The connection failed: Connection refused

估计则应该把localhost解析为172.17.0.1

1. 由于elasticsearch容器启动出现问题，同时该全docker都在一台机的架构又比较复杂，所以现在试图能否让ELK容器与以前搭过的PACS系统进行对接。由于刚才发现ELK的Kibana是可以在Docker host之外的Windows主机进行访问，所以对接应该能排除网络上问题，应该可以实现对接。

十四、关于elasticsearch的自动退出，如果对docker容器都进行重启，elasticsearch也是过一段时间会自动退出。

好像会出现这样的报错信息：

There is insufficient memory for the Java Runtime Environment to continue.

# Native memory allocation (mmap) failed to map 986513408 bytes for committing reserved memory.

有时则不是elasticsearch自动退出而是其它容器自动退出了，如果其它容器退出了那么elasticsearch就可能不退出。

解决办法：

修改/usr/share/elasticsearch/config/jvm.options

把

-Xms1g

-Xmx1g

改为：

-Xms512m

-Xmx512m

停掉所有的容器，在重新启动容器。然而并不能解决问题，因为其它容器也会自动退出。