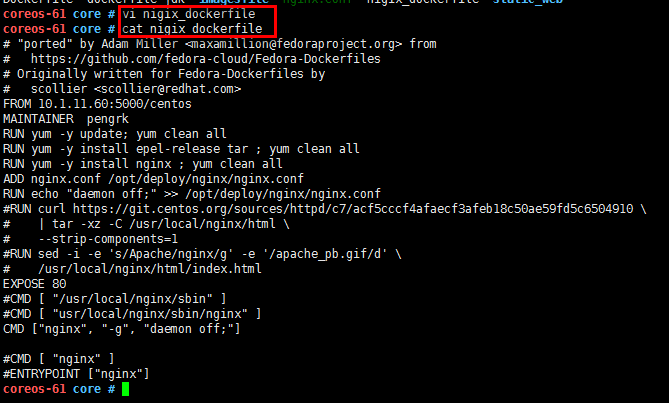
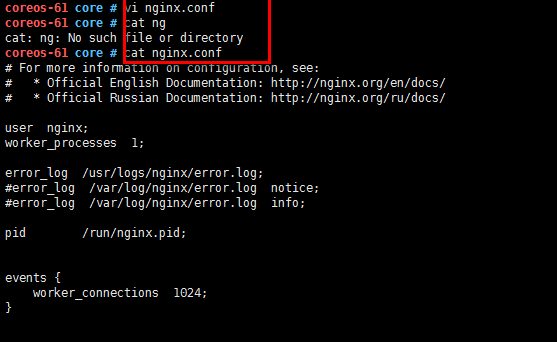
## DockerFile文件



|  |
| --- |
| # "ported" by Adam Miller <maxamillion@fedoraproject.org> from  # https://github.com/fedora-cloud/Fedora-Dockerfiles  # Originally written for Fedora-Dockerfiles by  # scollier <scollier@redhat.com>  FROM 10.1.11.60:5000/centos  MAINTAINER pengrk  RUN yum -y update; yum clean all  RUN yum -y install epel-release tar ; yum clean all  RUN yum -y install nginx ; yum clean all  ADD nginx.conf /opt/deploy/nginx/nginx.conf  RUN echo "daemon off;" >> /opt/deploy/nginx/nginx.conf  #RUN curl https://git.centos.org/sources/httpd/c7/acf5cccf4afaecf3afeb18c50ae59fd5c6504910 \  # | tar -xz -C /usr/local/nginx/html \  # --strip-components=1  #RUN sed -i -e 's/Apache/nginx/g' -e '/apache\_pb.gif/d' \  # /usr/local/nginx/html/index.html  EXPOSE 80  #CMD [ "/usr/local/nginx/sbin" ]  CMD ["nginx", "-g", "daemon off;"] |

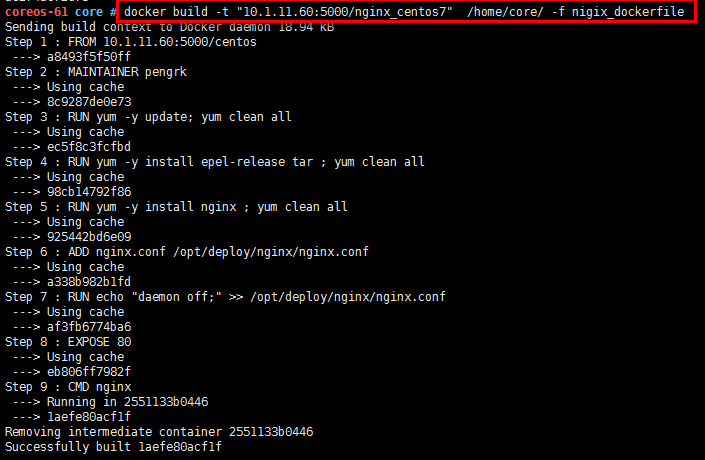
## nginx.conf



|  |
| --- |
| # For more information on configuration, see:  # \* Official English Documentation: http://nginx.org/en/docs/  # \* Official Russian Documentation: http://nginx.org/ru/docs/  user nginx;  worker\_processes 1;  error\_log /usr/logs/nginx/error.log;  #error\_log /var/log/nginx/error.log notice;  #error\_log /var/log/nginx/error.log info;  pid /run/nginx.pid;  events {  worker\_connections 1024;  }  http {  include mime.types;  default\_type application/octet-stream;  log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '  '$status $body\_bytes\_sent "$http\_referer" '  '"$http\_user\_agent" "$http\_x\_forwarded\_for"';  access\_log /usr/logs/nginx/access.log main;  sendfile on;  #tcp\_nopush on;  #keepalive\_timeout 0;  keepalive\_timeout 65;  #gzip on;  # Load modular configuration files from the /etc/nginx/conf.d directory.  # See http://nginx.org/en/docs/ngx\_core\_module.html#include  # for more information.  #include /etc/nginx/conf.d/\*.conf;  index index.html index.htm;  server {  listen 80;  server\_name localhost;  root /usr/share/nginx/html;  #charset koi8-r;  #access\_log /var/log/nginx/host.access.log main;  location / {  autoindex on;  }  # redirect server error pages to the static page /40x.html  #  error\_page 404 /404.html;  location = /40x.html {  }  # redirect server error pages to the static page /50x.html  #  error\_page 500 502 503 504 /50x.html;  location = /50x.html {  }  # proxy the PHP scripts to Apache listening on 127.0.0.1:80  #  #location ~ \.php$ {  # proxy\_pass http://127.0.0.1;  #}  # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000  #  #location ~ \.php$ {  # root html;  # fastcgi\_pass 127.0.0.1:9000;  # fastcgi\_index index.php;  # fastcgi\_param SCRIPT\_FILENAME /scripts$fastcgi\_script\_name;  # include fastcgi\_params;  #}  # deny access to .htaccess files, if Apache's document root  # concurs with nginx's one  #  #location ~ /\.ht {  # deny all;  #}  }  # another virtual host using mix of IP-, name-, and port-based configuration  #  #server {  # listen 8000;  # listen somename:8080;  # server\_name somename alias another.alias;  # root html;  # location / {  # }  #}  # HTTPS server  #  #server {  # listen 443;  # server\_name localhost;  # root html;  # ssl on;  # ssl\_certificate cert.pem;  # ssl\_certificate\_key cert.key;  # ssl\_session\_timeout 5m;  # ssl\_protocols SSLv2 SSLv3 TLSv1;  # ssl\_ciphers HIGH:!aNULL:!MD5;  # ssl\_prefer\_server\_ciphers on;  # location / {  # }  #}  } |

## build 镜像

docker build -t "10.1.11.60:5000/nginx\_centos7" /home/core/ -f nigix\_dockerfile



## run

docker run -i -t -d -p 81:80 10.1.11.60:5000/nginx\_centos7

curl <http://10.1.11.60:81>



## 参考：

http://blog.csdn.net/lk10207160511/article/details/51449762

1、资源准备

Dockerfile文件

**[plain]** [view plain](http://blog.csdn.net/lk10207160511/article/details/51449762) [copy](http://blog.csdn.net/lk10207160511/article/details/51449762)

[print?](http://blog.csdn.net/lk10207160511/article/details/51449762)[在CODE上查看代码片](https://code.csdn.net/snippets/1689969)

1. # "ported" by Adam Miller <maxamillion@fedoraproject.org> from
2. #   https://github.com/fedora-cloud/Fedora-Dockerfiles
3. #
4. # Originally written for Fedora-Dockerfiles by
5. #   scollier <scollier@redhat.com>
7. FROM centos:centos7
8. MAINTAINER The CentOS Project <cloud-ops@centos.org>
10. RUN yum -y update; yum clean all
11. RUN yum -y install epel-release tar ; yum clean all
12. RUN yum -y install nginx ; yum clean all
13. ADD nginx.conf /opt/deploy/nginx/nginx.conf
14. RUN echo "daemon off;" >> /opt/deploy/nginx/nginx.conf
15. #RUN curl https://git.centos.org/sources/httpd/c7/acf5cccf4afaecf3afeb18c50ae59fd5c6504910 \
16. #    | tar -xz -C /usr/local/nginx/html \
17. #    --strip-components=1
18. #RUN sed -i -e 's/Apache/nginx/g' -e '/apache\_pb.gif/d' \
19. #    /usr/local/nginx/html/index.html
21. EXPOSE 80
23. #CMD [ "/usr/local/nginx/sbin" ]

注意：路径需要在系统上面存在以及对应

nginx.conf文件

**[plain]** [view plain](http://blog.csdn.net/lk10207160511/article/details/51449762) [copy](http://blog.csdn.net/lk10207160511/article/details/51449762)

[print?](http://blog.csdn.net/lk10207160511/article/details/51449762)[在CODE上查看代码片](https://code.csdn.net/snippets/1689969)

1. # For more information on configuration, see:
2. #   \* Official English Documentation: http://nginx.org/en/docs/
3. #   \* Official Russian Documentation: http://nginx.org/ru/docs/
5. user  nginx;
6. worker\_processes  1;
8. error\_log  /usr/logs/nginx/error.log;
9. #error\_log  /var/log/nginx/error.log  notice;
10. #error\_log  /var/log/nginx/error.log  info;
12. pid        /run/nginx.pid;

15. events {
16. worker\_connections  1024;
17. }

20. http {
21. include       mime.types;
22. default\_type  application/octet-stream;
24. log\_format  main  '$remote\_addr - $remote\_user [$time\_local] "$request" '
25. '$status $body\_bytes\_sent "$http\_referer" '
26. '"$http\_user\_agent" "$http\_x\_forwarded\_for"';
28. access\_log  /usr/logs/nginx/access.log  main;
30. sendfile        on;
31. #tcp\_nopush     on;
33. #keepalive\_timeout  0;
34. keepalive\_timeout  65;
36. #gzip  on;
38. # Load modular configuration files from the /etc/nginx/conf.d directory.
39. # See http://nginx.org/en/docs/ngx\_core\_module.html#include
40. # for more information.
41. #include /etc/nginx/conf.d/\*.conf;
43. index   index.html index.htm;
45. server {
46. listen       80;
47. server\_name  localhost;
48. root         /usr/share/nginx/html;
50. #charset koi8-r;
52. #access\_log  /var/log/nginx/host.access.log  main;
54. location / {
55. autoindex on;
56. }
58. # redirect server error pages to the static page /40x.html
59. #
60. error\_page  404              /404.html;
61. location = /40x.html {
62. }
64. # redirect server error pages to the static page /50x.html
65. #
66. error\_page   500 502 503 504  /50x.html;
67. location = /50x.html {
68. }
70. # proxy the PHP scripts to Apache listening on 127.0.0.1:80
71. #
72. #location ~ \.php$ {
73. #    proxy\_pass   http://127.0.0.1;
74. #}
76. # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
77. #
78. #location ~ \.php$ {
79. #    root           html;
80. #    fastcgi\_pass   127.0.0.1:9000;
81. #    fastcgi\_index  index.php;
82. #    fastcgi\_param  SCRIPT\_FILENAME  /scripts$fastcgi\_script\_name;
83. #    include        fastcgi\_params;
84. #}
86. # deny access to .htaccess files, if Apache's document root
87. # concurs with nginx's one
88. #
89. #location ~ /\.ht {
90. #    deny  all;
91. #}
92. }

95. # another virtual host using mix of IP-, name-, and port-based configuration
96. #
97. #server {
98. #    listen       8000;
99. #    listen       somename:8080;
100. #    server\_name  somename  alias  another.alias;
101. #    root         html;
103. #    location / {
104. #    }
105. #}

108. # HTTPS server
109. #
110. #server {
111. #    listen       443;
112. #    server\_name  localhost;
113. #    root         html;
115. #    ssl                  on;
116. #    ssl\_certificate      cert.pem;
117. #    ssl\_certificate\_key  cert.key;
119. #    ssl\_session\_timeout  5m;
121. #    ssl\_protocols  SSLv2 SSLv3 TLSv1;
122. #    ssl\_ciphers  HIGH:!aNULL:!MD5;
123. #    ssl\_prefer\_server\_ciphers   on;
125. #    location / {
126. #    }
127. #}
129. }

注意：路径需要在系统上面存在以及对应

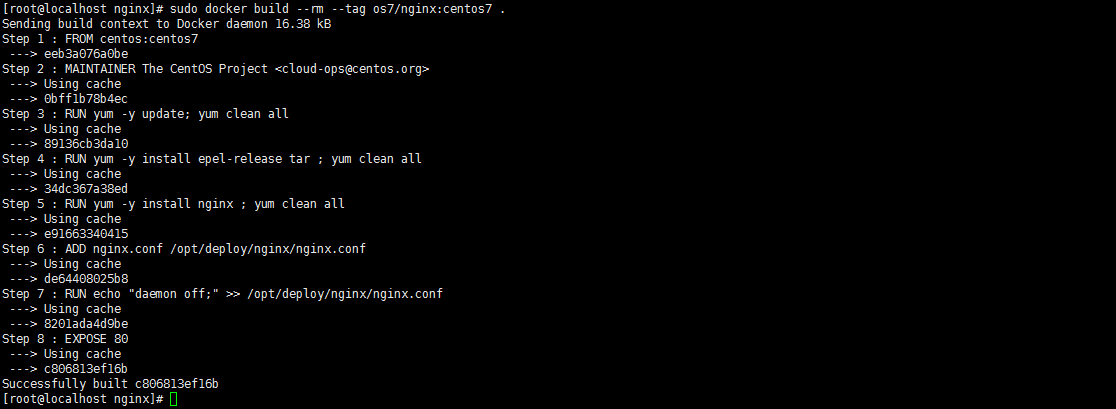
2、执行构建镜像命令

**[plain]** [view plain](http://blog.csdn.net/lk10207160511/article/details/51449762) [copy](http://blog.csdn.net/lk10207160511/article/details/51449762)

[print?](http://blog.csdn.net/lk10207160511/article/details/51449762)[在CODE上查看代码片](https://code.csdn.net/snippets/1689969)

1. [root@localhost nginx]# sudo docker build --rm --tag os7/nginx:centos7 .

执行结果截图：



3、查看镜像是否安装构建成功 **[Docker](http://lib.csdn.net/base/docker" \o "Docker知识库" \t "_blank)** images

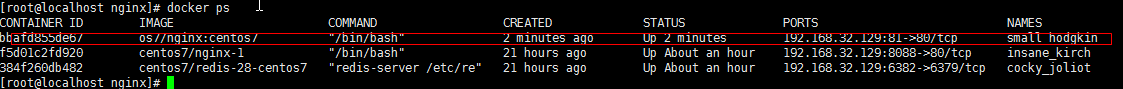


4、创建容器 docker run -i -t -d -p 192.168.32.129:81:80 os7/nginx /bin/bash

注意：192.168.32.129这个IP的话，则需要在/etc/hosts中添加

192.168.32.129     localhost

5、查看容器是否创建成功并启动 docker ps



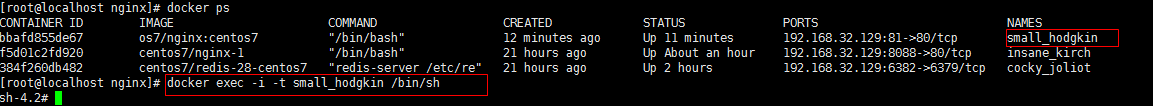
6、**[测试](http://lib.csdn.net/base/softwaretest" \o "软件测试知识库" \t "_blank)**是否成功访问 curl http://192.168.32.129:81

http://img.blog.csdn.net/20160519230452859

会出现这个拒绝连接，那怎么办呢？有办法解决的，我们先进入该容器里面

下面红色换成自己的container\_id，如果指定了名称，用名称也可以

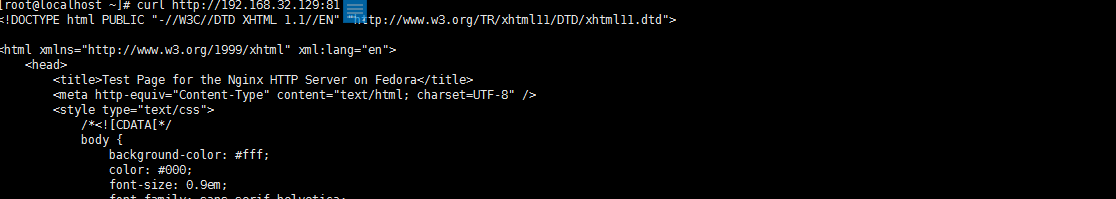
7、进入容器 docker exec -i -t small\_hodgkin /bin/sh



8、接着在容器里面执行（直接输入即可）

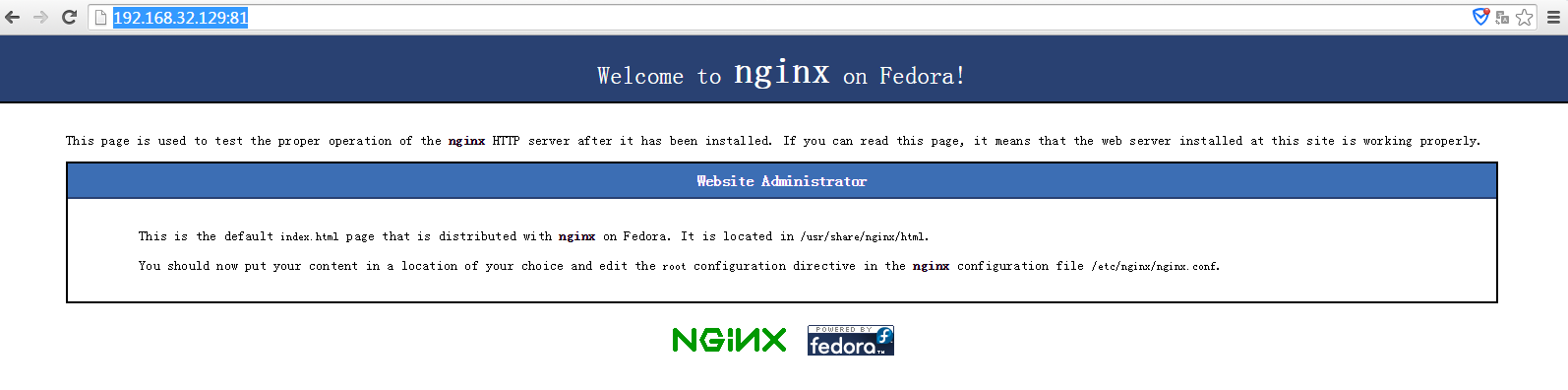
nginx

9、在容器外面执行 curl http://192.168.32.129:81



成功了。

10、再到虚拟机外面通过浏览器访问



到此为止就成功了。

参考资料：

<https://github.com/CentOS/CentOS-Dockerfiles>

## 参考2：

FROM debian:jessie  
  
MAINTAINER NGINX Docker Maintainers "docker-maint@nginx.com"  
  
ENV NGINX\_VERSION 1.11.13-1~jessie  
  
RUN set -e; \  
NGINX\_GPGKEY=573BFD6B3D8FBC641079A6ABABF5BD827BD9BF62; \  
found=''; \  
for server in \  
C:\Users\prk\AppData\Local\Temp\%W@GJ$ACOF(TYDYECOKVDYB.pngha.pool.sks-keyservers.net \  
hkp://C:\Users\prk\AppData\Local\Temp\%W@GJ$ACOF(TYDYECOKVDYB.pngkeyserver.ubuntu.com:80 \  
hkp://C:\Users\prk\AppData\Local\Temp\%W@GJ$ACOF(TYDYECOKVDYB.pngp80.pool.sks-keyservers.net:80 \  
C:\Users\prk\AppData\Local\Temp\[5UQ[BL(6~BS2JV6W}N6[%S.pngpgp.mit.edu \  
; do \  
echo "Fetching GPG key $NGINX\_GPGKEY from $server"; \  
apt-key adv --keyserver "$server" --keyserver-options timeout=10 --recv-keys "$NGINX\_GPGKEY" && found=yes && break; \  
done; \  
test -z "$found" && echo >&2 "error: failed to fetch GPG key $NGINX\_GPGKEY" && exit 1; \  
exit 0  
  
RUN echo "deb C:\Users\prk\AppData\Local\Temp\%W@GJ$ACOF(TYDYECOKVDYB.pnghttp://nginx.org/packages/mainline/debian/ jessie nginx" >> /etc/apt/sources.list \  
&& apt-get update \  
&& apt-get install --no-install-recommends --no-install-suggests -y \  
ca-certificates \  
nginx=${NGINX\_VERSION} \  
nginx-module-xslt \  
nginx-module-geoip \  
nginx-module-image-filter \  
nginx-module-perl \  
nginx-module-njs \  
gettext-base \  
&& rm -rf /var/lib/apt/lists/\*  
  
# forward request and error logs to docker log collector  
RUN ln -sf /dev/stdout /var/log/nginx/access.log \  
&& ln -sf /dev/stderr /var/log/nginx/error.log  
  
EXPOSE 80 443  
  
CMD ["nginx", "-g", "daemon off;"]