南昌大学实验报告

姓名：**陈华豪**

学号：**6130116238**

邮箱地址：**6130116238@email.ncu.edu.cn**

专业班级：**网络工程161班**

实验日期：**2019.4.21**

课程名称：**网络协议分析与实现**

实验项目名称

Lab 3: Live Migration

实验目的

- Understanding the basic techniques for VM migration
- Using docker container as an example to test your migration skill
- Understanding the concept of checkpoint and restore
- Successfully migrate your image from one host to another
- Writing a decent report

实验基础

- Learn to do live migration in docker
- Using Checkpoint and Restore (CRIU) tool in docker to do this job
- https://criu.org/Live_migration is your source to check
- Ask one of our friend/roommate and migrate your Golang job (with your ID or name) to his/her place
- Measure the duration of migration and collect all other runtime statistics

实验步骤

代码步骤：

> 1：echo "{\"experimental\": true}" >> /etc/docker/daemon.json
>
> systemctl restart docker
>
> 2：docker run -d --name looper --security-opt seccomp:unconfined busybox \
>
> /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; done'
>
> 3：docker logs looper
>
> 4：docker checkpoint create looper checkpoint1
>
> 5：docker start --checkpoint checkpoint1 looper
>
> 6：docker run -d --name looper2 --security-opt seccomp:unconfined busybox \
>
> /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; done'
>
> 7：docker checkpoint create --checkpoint-dir=/tmp looper2 checkpoint2
>
> 8：docker create --name looper-clone --security-opt seccomp:unconfined busybox \
>
> /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; done'
>
> 9：docker start --checkpoint-dir=/tmp --checkpoint=checkpoint2 looper-clone
>
> 10：sudo scp -r golang.tar root@159.65.70.253

截图：


```
kproot@vps-sfo181022:~# echo "{\"experimental\": true}" >> /etc/docker/daemon.json
--root@vps-sfo181022:~# systemctl restart docker
s aroot@vps-sfo181022:~# 
```

point. These checkpoints are stored and managed by Docker, unless you specify a custom storage

an example of creating a checkpoint, from a container that simply logs an integer in a loop.


```
774MB
golang              <none>              c7942203692b        2 weeks ago
You can 774MB
root@vps-sfo181022:~# echo "{\"experimental\": true}" >> /etc/docker/daemon.json
root@vps-sfo181022:~# systemctl restart docker
$ dock root@vps-sfo181022:~# docker run -d --name looper --security-opt seccomp:unconfi
ned busybox  \
>          /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; d
one'
If you doUnable to find image 'busybox:latest' locally
latest: Pulling from library/busybox
fc1a6b909f82: Pull complete
$ dock Digest: sha256:954e1f01e80ce09d0887ff6ea10b13a812cb01932a0781d6b0cc23f743a874fd
Status: Downloaded newer image for busybox:latest
dde069d75149e266e20ddb43ed7518c6d319956544b915a3f07fc0af8b6853e6
You sho root@vps-sfo181022:~# docker logs looper
0
1
2
restore 3
4
Unlike c 5
example 6
7
```

```
$ docker start --checkpoint checkpoint1 looper
```

You can

$ dock

If you do

$ dock

You sho

restore

Unlike d
example

```
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
root@vps-sfo181022:~# docker checkpoint create looper checkpoint1
checkpoint1
root@vps-sfo181022:~#
```

$ dock

```
root@vps-sfo181022:~# docker run -d --name looper2 --security-opt seccomp:unconf
ined busybox \
>          /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; d
one'
f331f0bad02caea6d8d08c6a4fb07dbf707d7a57468d5585b125f1940dffe56f
root@vps-sfo181022:~#
```

ou should be able to print the logs from `looper-clone` and see that they start from wherever the logs of

Passing additional options [edit]

Restorin

Beyond

containe

path wit

$ dock

wait

$ dock

$ dock

$ dock

```
17
18
19
20
21
22
23
24
25
26
27
28
root@vps-sfo181022:~# docker checkpoint create looper checkpoint1
checkpoint1
root@vps-sfo181022:~# docker start --checkpoint checkpoint1 looper
root@vps-sfo181022:~# docker run -d --name looper2 --security-opt seccomp:unconf
ined busybox \
>          /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; d
one'
f331f0bad02caea6d8d08c6a4fb07dbf707d7a57468d5585b125f1940dffe56f
root@vps-sfo181022:~# docker checkpoint create --checkpoint-dir=/tmp looper2 che
ckpoint2
checkpoint2
root@vps-sfo181022:~#
```

You should be able to print the logs from `looper-clone` and see that they start from wherever the logs of

Passing additional options [edit]

Restorir

Beyond

containe

path wit

```
22
23
24
25
26
27
28
root@vps-sfo181022:~# docker checkpoint create looper checkpoint1
checkpoint1
root@vps-sfo181022:~# docker start --checkpoint checkpoint1 looper
root@vps-sfo181022:~# docker run -d --name looper2 --security-opt seccomp:unconf
ined busybox \
>          /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; d
one'
f331f0bad02caea6d8d08c6a4fb07dbf707d7a57468d5585b125f1940dffe56f
root@vps-sfo181022:~# docker checkpoint create --checkpoint-dir=/tmp looper2 che
ckpoint2
checkpoint2
root@vps-sfo181022:~# docker create --name looper-clone --security-opt seccomp:u
nconfined busybox \
>          /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; d
one'
c2da7f58da8d58095c5c796a312353dfb7493ba88f8c7acbdfe25c5ed5faece2
root@vps-sfo181022:~#
```

$ dock

wait
one'
$ dock

$ dock

$ dock

You should be able to print the logs from `looper-clone` and see that they start from wherever the logs of

Passing additional options [edit]

```
root@vps-sfo181022:~# docker start --checkpoint-dir=/tmp --checkpoint=checkpoint
2 looper-clone
Error response from daemon: custom checkpointdir is not supported
root@vps-sfo181022:~#
```

creating a checkpoint, restoring from a checkpoint is just a flag provided to the normal container st
ble:

Restorir

Beyond

containe

path wit

```
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
root@vps-sfo181022:~#
```

$ dock

# wait
$ dock

$ dock

$ dock

You should be able to print the logs from `looper-clone` and see that they start from wherever the logs of

Passing additional options [edit]

Restorin

Beyond

containe

path wit

```
$ dock
# wait
$ dock
$ dock
$ dock
```

```
ined busybox \
>         /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; d
one'
f331f0bad02caea6d8d08c6a4fb07dbf707d7a57468d5585b125f1940dffe56f
root@vps-sfo181022:~# docker checkpoint create --checkpoint-dir=/tmp looper2 che
ckpoint2
checkpoint2
root@vps-sfo181022:~# docker create --name looper-clone --security-opt seccomp:u
nconfined busybox \
>         /bin/sh -c 'i=0; while true; do echo $i; i=$(expr $i + 1); sleep 1; d
one'
c2da7f58da8d58095c5c796a312353dfb7493ba88f8c7acbdfe25c5ed5faece2
root@vps-sfo181022:~# docker start --checkpoint-dir=/tmp --checkpoint=checkpoint
2 looper-clone
Error response from daemon: custom checkpointdir is not supported
root@vps-sfo181022:~# docker checkpoint create looper checkpoint2
checkpoint2
root@vps-sfo181022:~# docker logs looper
0
1
2
3
4
5
```

You should be able to print the logs from `looper-clone` and see that they start from wherever the logs of

Passing additional options [edit]

ools

/hat link

elated c

pecial pages

rintable version

ermanent link

```
shadowsocks-all.sh    tee                    wordpress-4.8-zh_CN.zip
root@vps-sfo181022:~# cd /home
root@vps-sfo181022:/home# ls
cogito0   golang.tar
root@vps-sfo181022:/home# 
```

2 Compatibility Notes

   2.1 TTY

```
root@yuehao-virtual-machine:/home/yuehao# ssh 159.65.70.253
The authenticity of host '159.65.70.253 (159.65.70.253)' can't be establishe
ECDSA key fingerprint is SHA256:UYV+/PzAkC45w4Ze3qXYTp7lmKbCz7fmcK6/zKr4VAY.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '159.65.70.253' (ECDSA) to the list of known host
root@159.65.70.253's password:
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-36-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sun Apr 21 15:05:47 UTC 2019

  System load:  0.0                Processes:             109
  Usage of /:   18.0% of 24.06GB   Users logged in:       1
  Memory usage: 54%                IP address for eth0:   159.65.70.253
  Swap usage:   0%                 IP address for docker0: 172.17.0.1

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud
```

实验数据或结果

结果如上。

实验思考

参考资料

- https://criu.org/Docker