To Kill or to Checkpoint - That is the Question

Mike Rapoport
<rppt@linux.vnet.ibm.com>

Adrian Reber
<areber@redhat.com>

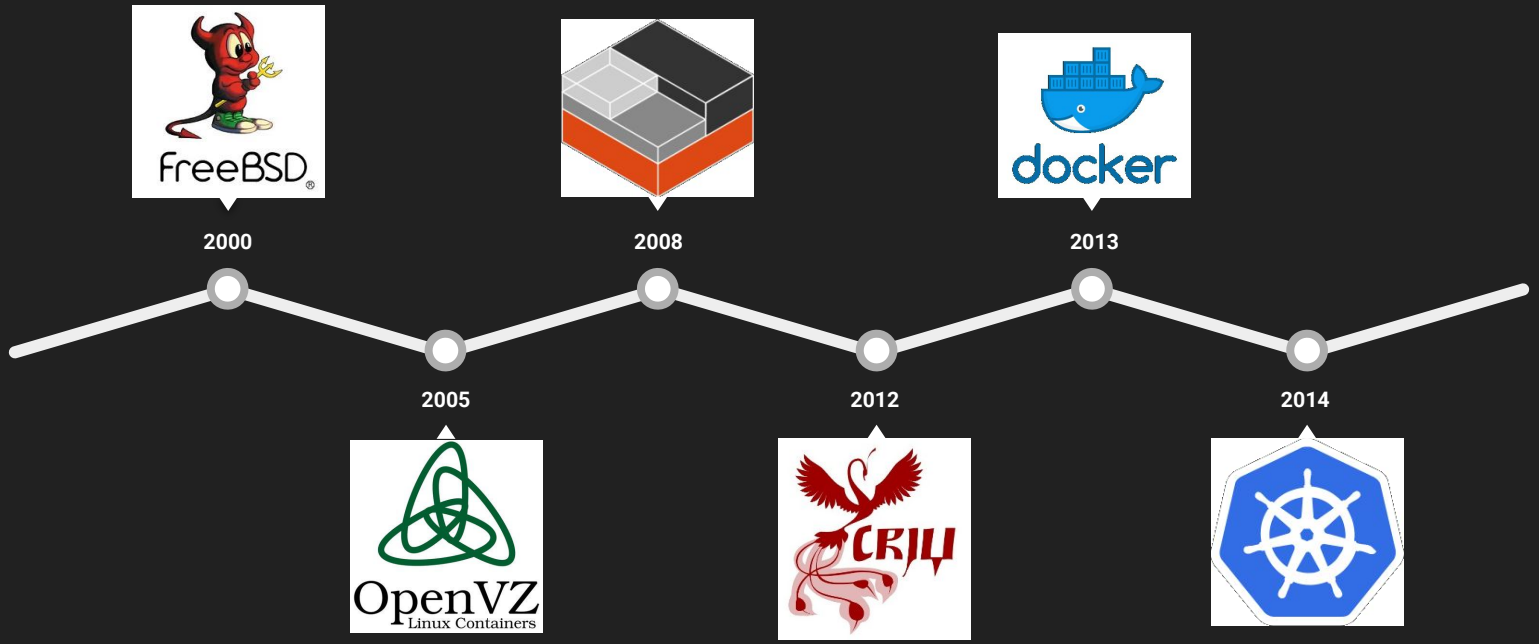
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 688386

Cloud native era

- Applications are
  - Adaptable
  - Resilient
  - Stateless

- No need for migration, just kill here and start there

History


FreeBSD
2000
2005
OpenVZ
Linux Containers
2008
2012
CRIU
docker
2013
2014

CRIU today

- Supports several architectures
- Integrated with major container engines
  - Docker
  - LXC/D
  - OpenVZ
  - podman
  - runc
- Used in production
  - IBM Spectrum LSF
  - Google
  - Virtuozzo

Simple example

- Shell script
  - print the date each second

```
#!/bin/sh

while :; do
    sleep 1
    date
done
```

- https://criu.org/Simple_loop
- https://asciinema.org/a/c5j7RTcYkcQqFGpsqiiuIkoaj

Simple example

```
criu@criu-dev:~/examples/simple_loop
$ ls
loop.sh
criu@criu-dev:~/examples/simple_loop
$ cat ./loop.sh
#!/bin/sh

while :; do
    sleep 1
    date
done
criu@criu-dev:~/examples/simple_loop
$ ./loop.sh
Thu Oct 18 08:35:49 UTC 2018
Thu Oct 18 08:35:50 UTC 2018
Thu Oct 18 08:35:51 UTC 2018
Thu Oct 18 08:35:52 UTC 2018
Thu Oct 18 08:35:53 UTC 2018
Thu Oct 18 08:35:54 UTC 2018
Thu Oct 18 08:35:55 UTC 2018
Thu Oct 18 08:35:56 UTC 2018
Thu Oct 18 08:35:57 UTC 2018
Thu Oct 18 08:35:58 UTC 2018
Thu Oct 18 08:35:59 UTC 2018
Thu Oct 18 08:36:00 UTC 2018
Thu Oct 18 08:36:01 UTC 2018
Thu Oct 18 08:36:02 UTC 2018
Thu Oct 18 08:36:03 UTC 2018
Thu Oct 18 08:36:04 UTC 2018
Thu Oct 18 08:36:05 UTC 2018
Thu Oct 18 08:36:06 UTC 2018
Thu Oct 18 08:36:07 UTC 2018
Thu Oct 18 08:36:08 UTC 2018
Thu Oct 18 08:36:09 UTC 2018
Thu Oct 18 08:36:10 UTC 2018
Killed
criu@criu-dev:~/examples/simple_loop
$
```

```
criu@criu-dev:~/dumps/simple_loop
$ sudo criu dump -v0 --shell-job -t $(pidof -x loop.sh) && echo OK
OK
criu@criu-dev:~/dumps/simple_loop
$ sudo criu restore -v0 --shell-job
Thu Oct 18 08:36:25 UTC 2018
Thu Oct 18 08:36:26 UTC 2018
Thu Oct 18 08:36:27 UTC 2018
Thu Oct 18 08:36:28 UTC 2018
Thu Oct 18 08:36:29 UTC 2018
Thu Oct 18 08:36:30 UTC 2018
```

```
[0] 0:sudo*                                                    "criu-dev" 08:36 18-Oct-18
```

Live TCP

- TCP echo server and client

```c
/* client */

int val = 1;

while (1) {
        write(sk, &val, sizeof(val));
        read(sk, &rval, sizeof(rval));
        printf("PP %d -> %d\n", val, rval);
        sleep(2);
        val++;
}
```

```c
/* server */

int val;

while (1) {
        read(sk, &val, sizeof(val));
        write(sk, &val, sizeof(val));
}
```

- https://criu.org/Simple_TCP_pair
- https://asciinema.org/a/yjC1IK4S0ZG8Dz1UFI3CK0Vnc

Live TCP

```
PP 5 -> 5                                  criu@criu-dev:~/dumps/tcp_pair
PP 6 -> 6                                  $ sudo criu dump -v0 --shell-job --tcp-established -t $(pidof client) && echo OK
PP 7 -> 7                                  OK
PP 8 -> 8                                  criu@criu-dev:~/dumps/tcp_pair
PP 9 -> 9                                  $ sudo criu restore -v0 --shell-job --tcp-established
PP 10 -> 10                                PP 22 -> 22
PP 11 -> 11                                PP 23 -> 23
PP 12 -> 12                                
PP 13 -> 13
PP 14 -> 14
PP 15 -> 15
PP 16 -> 16
PP 17 -> 17
PP 18 -> 18
PP 19 -> 19
PP 20 -> 20
PP 21 -> 21
Killed
criu@criu-dev:~/examples/tcp_pair
$
```

```
criu@criu-dev:~/examples/tcp_pair          09:17:24.164352 IP 127.0.0.1.9876 > 127.0.0.1.53252: tcp 4
$ ./server 9876                            09:17:24.164364 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 0
Binding to port 9876                       09:17:26.164651 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 4
Waiting for connections                    09:17:26.164731 IP 127.0.0.1.9876 > 127.0.0.1.53252: tcp 4
New connection                             09:17:26.164742 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 0
```

```
                                           09:17:42.220738 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 0
                                           09:17:42.220749 IP 127.0.0.1.9876 > 127.0.0.1.53252: tcp 0
                                           09:17:42.221147 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 4
                                           09:17:42.221214 IP 127.0.0.1.9876 > 127.0.0.1.53252: tcp 4
                                           09:17:42.221240 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 0
                                           09:17:44.221490 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 4
                                           09:17:44.221562 IP 127.0.0.1.9876 > 127.0.0.1.53252: tcp 4
                                           09:17:44.221584 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 0
[0] 0:sudo*                                                                  "criu-dev" 09:17 18-Oct-18
```

Iterative dump

- Checkpoint - restore of memcached server
  - Several pre-dump iterations
  - Final dump and server freeze
  - Lazy restore
- https://asciinema.org/a/4olkCRLaMGa4GhxTEvbzyTtv4

Iterative dump

```
GET: 17: 100 values                 criu@criu-dev:~/dumps/memcached
GET: 18: 100 values                 $ mkdir 1 2 3 4
GET: 19: 100 values                 criu@criu-dev:~/dumps/memcached
GET: 20: 100 values                 $ sudo criu pre-dump --images-dir 1 -t $(pidof memcached) --shell-job --tcp-established -v0 &&
GET: 21: 100 values                 echo OK
GET: 22: 100 values                 OK
GET: 23: 100 values                 criu@criu-dev:~/dumps/memcached
GET: 24: 100 values                 $ sudo criu pre-dump --images-dir 2 --prev-images-dir ../1 --track-mem -t $(pidof memcached) --
GET: 25: 100 values                 shell-job --tcp-established -v0 && echo OK
GET: 26: 100 values                 OK
GET: 27: 100 values                 criu@criu-dev:~/dumps/memcached
GET: 28: 100 values                 $ ls -l */pages*
GET: 29: 100 values                 -rw-r--r-- 1 root root  475136 Oct 18 09:52 1/pages-1.img
Putting 100 more values             -rw-r--r-- 1 root root 1642496 Oct 18 09:52 2/pages-1.img
GET: 0: 200 values                  criu@criu-dev:~/dumps/memcached
GET: 1: 200 values                  $ sudo criu pre-dump --images-dir 3 --prev-images-dir ../2 --track-mem -t $(pidof memcached) --
GET: 2: 200 values                  shell-job --tcp-established -v0 && echo OK
GET: 3: 200 values                  OK
GET: 4: 200 values                  criu@criu-dev:~/dumps/memcached
GET: 5: 200 values                  $ ls -l */pages*
GET: 6: 200 values                  -rw-r--r-- 1 root root  475136 Oct 18 09:52 1/pages-1.img
GET: 7: 200 values                  -rw-r--r-- 1 root root 1642496 Oct 18 09:52 2/pages-1.img
GET: 8: 200 values                  -rw-r--r-- 1 root root  614400 Oct 18 09:53 3/pages-1.img
GET: 9: 200 values                  criu@criu-dev:~/dumps/memcached
GET: 10: 200 values                 $ sudo criu dump --images-dir 4 --prev-images-dir ../3 --track-mem -t $(pidof memcached) --shel
GET: 11: 200 values                 l-job --tcp-established -v0 && echo OK
GET: 12: 200 values                 OK
GET: 13: 200 values                 criu@criu-dev:~/dumps/memcached
GET: 14: 200 values                 $ sudo criu lazy-pages --images-dir 4 -v0 &
GET: 15: 200 values                 [1] 24259
GET: 16: 200 values                 criu@criu-dev:~/dumps/memcached
GET: 17: 200 values                 $ sudo criu restore --shell-job --tcp-established --images-dir 4 --lazy-pages -d  && echo OK
GET: 18: 200 values                 [1]+  Done                    sudo criu lazy-pages --images-dir 4 -v0
GET: 19: 200 values                 OK
GET: 20: 200 values                 criu@criu-dev:~/dumps/memcached
GET: 21: 200 values                 $
GET: 22: 200 values
GET: 23: 200 values
GET: 24: 200 values
[0] 0:bash*                                                                                  "criu-dev" 09:53 18-Oct-18
```

Iterative dump with LXD

- Iterative container migration using LXD
  - First migration threshold: 90%
  - Second migration threshold: 40%
  - Different number of pre-copy runs depending on these thresholds
- https://asciinema.org/a/mUxKQHnPbRLdNMNukE2WuJku7
  - Hosts: RHEL7
  - Container: Alpine 3.7

Iterative dump with LXD

```
| NAME | STATE | IPV4 | IPV6 | TYPE | SNAPSHOTS |
+------+-------+------+------+------+-----------+
[root@rhel01 ~]# lxc move alpine rhel02:alpine
[root@rhel01 ~]# lxc list
+------+-------+------+------+------+-----------+
| NAME | STATE | IPV4 | IPV6 | TYPE | SNAPSHOTS |
+------+-------+------+------+------+-----------+
[root@rhel01 ~]# lxc list rhel02:
+--------+---------+------+------+------------+-----------+
|  NAME  |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+--------+---------+------+------+------------+-----------+
| alpine | RUNNING |      |      | PERSISTENT |           |
+--------+---------+------+------+------------+-----------+
[root@rhel01 ~]# lxc mv rhel02:alpine alpine
[root@rhel01 ~]# lxc list
+--------+---------+------+------+------------+-----------+
|  NAME  |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+--------+---------+------+------+------------+-----------+
| alpine | RUNNING |      |      | PERSISTENT |           |
+--------+---------+------+------+------------+-----------+
[root@rhel01 ~]# lxc list
```

0 adrian@dcbz:~

```
CRIU pages skipped 0
CRIU pages skipped percentage 0%
Doing another pre-dump in 001
CRIU pages written 42049
CRIU pages skipped 60408
CRIU pages skipped percentage 59%
Doing another pre-dump in 003
CRIU pages written 18750
CRIU pages skipped 83707
CRIU pages skipped percentage 82%
Doing another pre-dump in 005
CRIU pages written 10880
CRIU pages skipped 91577
CRIU pages skipped percentage 90%
Doing another pre-dump in 007
CRIU pages written 8043
CRIU pages skipped 94414
CRIU pages skipped percentage 93%
Memory pages skipped (93%) due to pre-copy is larger than threshold (90%)
This was the last pre-dump; next dump is the final dump
```

1 adrian@dcbz:~

```
Using maximal 10 iterations for pre-dumping
The other side does support pre-copy
Doing another pre-dump in
CRIU pages written 102457
CRIU pages skipped 0
CRIU pages skipped percentage 0%
Doing another pre-dump in 001
CRIU pages written 55897
CRIU pages skipped 46560
CRIU pages skipped percentage 46%
Memory pages skipped (46%) due to pre-copy is larger than threshold (40%)
This was the last pre-dump; next dump is the final dump
```

3 root@rhel02:~

Application snapshot

- Checkpoint/restore using Podman
  - Apache Tomcat running in a container
  - Container is checkpointed
  - System rebooted
  - Container is restored
- https://asciinema.org/a/FsTbx9mZkzeuhCM2pFOr1tujM

Application snapshot with podman

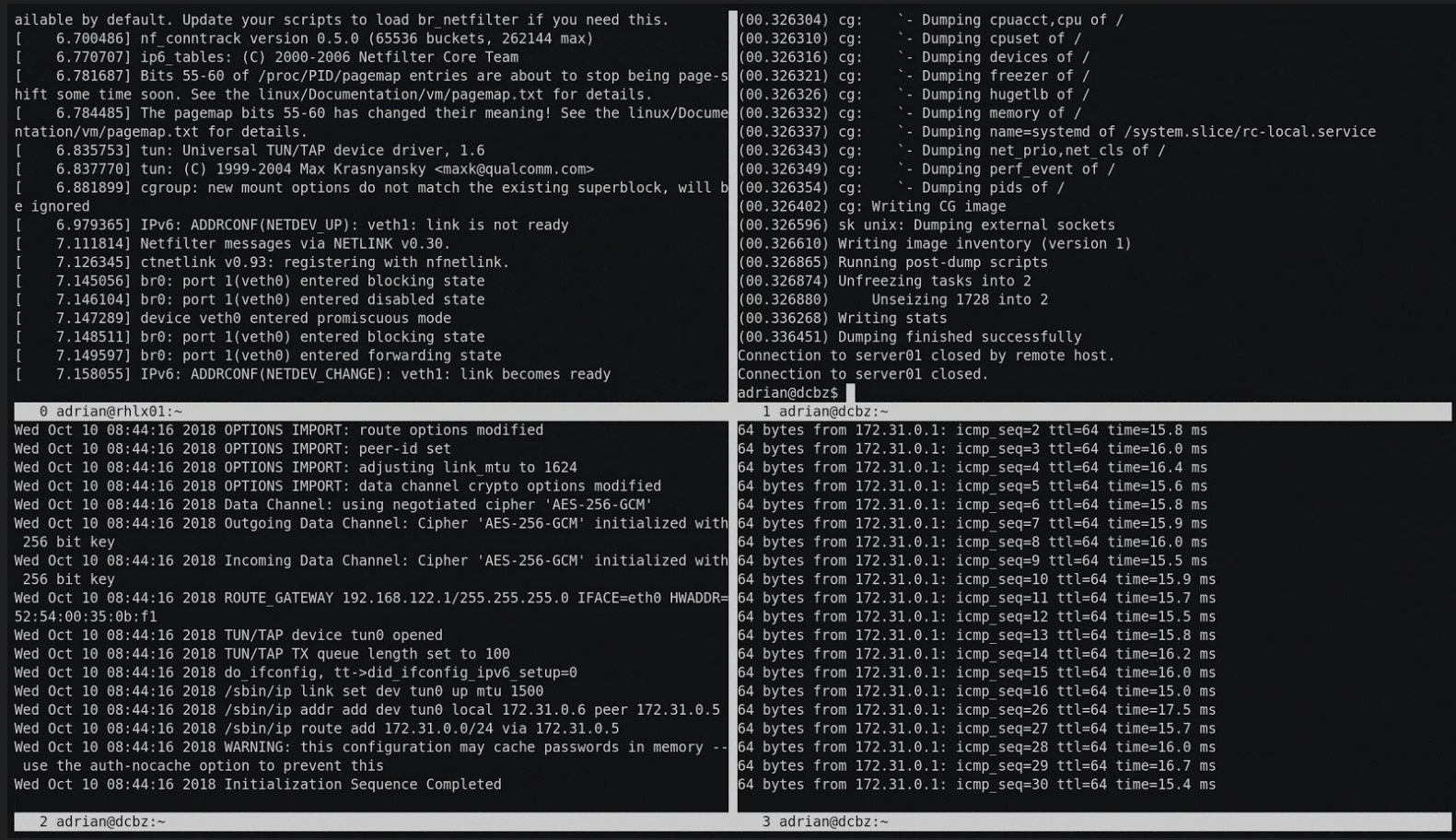
```
Connection to rhel01 closed.
[root@rhel02 ~]# ssh rhel01
✔  ~
root@rhel01  #  podman container restore --keep podman-criu-test
9f4b568a05e598297763ebc6bbcebbede351ab781d55f3a7a792e42b95e01a80
✔  ~
root@rhel01  #  reboot
PolicyKit daemon disconnected from the bus.
We are no longer a registered authentication agent.
Connection to rhel01 closed by remote host.
Connection to rhel01 closed.
[root@rhel02 ~]# ssh rhel01
✔  ~
root@rhel01  #  podman container restore --keep podman-criu-test
9f4b568a05e598297763ebc6bbcebbede351ab781d55f3a7a792e42b95e01a80
✔  ~
root@rhel01  #
  0 root@rhel02:~
[root@rhel02 ~]# curl 10.22.0.53:8080/examples/servlets/servlet/HelloWorldExample
1
[root@rhel02 ~]# curl 10.22.0.53:8080/examples/servlets/servlet/HelloWorldExample
2
[root@rhel02 ~]#  curl 10.22.0.53:8080/examples/servlets/servlet/HelloWorldExample
3
[root@rhel02 ~]# curl 10.22.0.53:8080/examples/servlets/servlet/HelloWorldExample
4
[root@rhel02 ~]# curl 10.22.0.53:8080/examples/servlets/servlet/HelloWorldExample
5
[root@rhel02 ~]#  curl 10.22.0.53:8080/examples/servlets/servlet/HelloWorldExample
3
[root@rhel02 ~]# curl 10.22.0.53:8080/examples/servlets/servlet/HelloWorldExample
4
[root@rhel02 ~]#


  1 root@rhel02:~
```

CRIU and kexec

- Checkpoint OpenVPN server
  - While client connected via TCP
- kexec into different kernel
- Restore OpenVPN server
  - Client still connected
- https://asciinema.org/a/i6PF2bJo79QP6pnhqJwX4Wfs2

CRIU and kexec



Serverless

- Openwhisk action runtimes
  - Python
  - Java
  - Node.js

`docker run` vs `docker start --checkpoint`

	python3action	nodejs6action	java8action
Start time (sec)	0.919969	0.743099	0.566213
Restore time (sec)	0.078663	0.091419	0.089104

Data store optimizations

- Populate vs migrate
  - Store with 100000 short entries

	memcached	redis
Populate time (sec)	1.238	6.254
Migrate time (sec)	0.806	1.671

- C/R and migration help:
  - Keep the caches
  - Reduce footprint for master-slave configuration
  - Run redis with transparent huge pages ;-)

More use cases

- Application forensics
  - Get application state from production
  - Investigate misbehaviour offline
- Fault tolerant and highly available systems
  - Use CRIU as state replication engine
- Load balancing
- Move applications into **`screen/tmux`**

Future: CRIU and kubernetes

- **`kubectl drain --migrate`**
- k8s replica sets with restore:

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: datastore
spec:
  replicas: 5
  startOptions:
    - checkpointRestore
```

Thank you!