

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

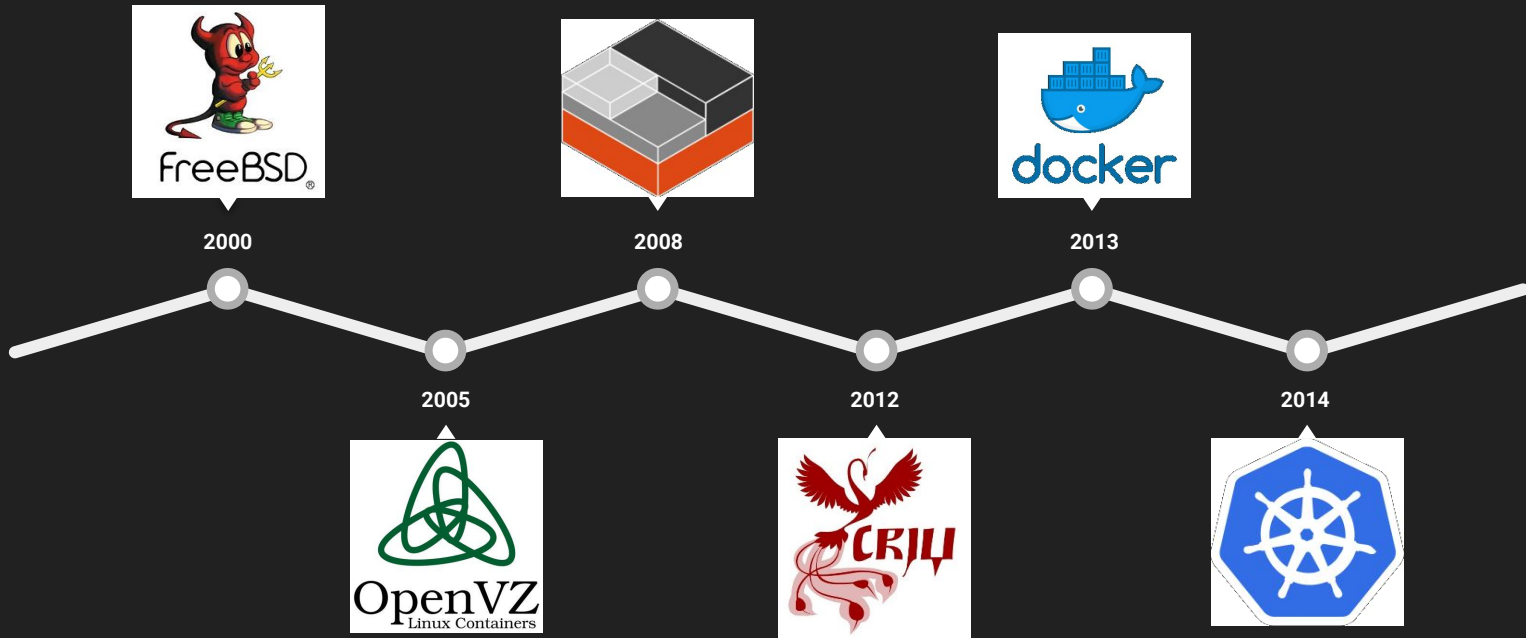


Cloud native era

- Applications are
 - Adaptable
 - Resilient
 - Stateless
- No need for migration, just kill here and start there



History



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/c5j7RTcYkcQqFGpsqiiulkoaj>

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
```

█

Live TCP

- TCP echo server and client

```
/* 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++;
}
```

```
/* 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
PP 6 -> 6
PP 7 -> 7
PP 8 -> 8
PP 9 -> 9
PP 10 -> 10
PP 11 -> 11
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
$ ./server 9876
Binding to port 9876
Waiting for connections
New connection

criu@criu-dev:~/dumps/tcp_pair
$ sudo criu dump -v0 --shell-job --tcp-established -t $(pidof client) && echo OK
OK
criu@criu-dev:~/dumps/tcp_pair
$ sudo criu restore -v0 --shell-job --tcp-established
PP 22 -> 22
PP 23 -> 23
█

09:17:24.164352 IP 127.0.0.1.9876 > 127.0.0.1.53252: tcp 4
09:17:24.164364 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 0
09:17:26.164651 IP 127.0.0.1.53252 > 127.0.0.1.9876: tcp 4
09:17:26.164731 IP 127.0.0.1.9876 > 127.0.0.1.53252: tcp 4
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
GET: 18: 100 values
GET: 19: 100 values
GET: 20: 100 values
GET: 21: 100 values
GET: 22: 100 values
GET: 23: 100 values
GET: 24: 100 values
GET: 25: 100 values
GET: 26: 100 values
GET: 27: 100 values
GET: 28: 100 values
GET: 29: 100 values
Putting 100 more values
GET: 0: 200 values
GET: 1: 200 values
GET: 2: 200 values
GET: 3: 200 values
GET: 4: 200 values
GET: 5: 200 values
GET: 6: 200 values
GET: 7: 200 values
GET: 8: 200 values
GET: 9: 200 values
GET: 10: 200 values
GET: 11: 200 values
GET: 12: 200 values
GET: 13: 200 values
GET: 14: 200 values
GET: 15: 200 values
GET: 16: 200 values
GET: 17: 200 values
GET: 18: 200 values
GET: 19: 200 values
GET: 20: 200 values
GET: 21: 200 values
GET: 22: 200 values
GET: 23: 200 values
GET: 24: 200 values
```

```
criu@criu-dev:~/dumps/memcached
$ mkdir 1 2 3 4
criu@criu-dev:~/dumps/memcached
$ sudo criu pre-dump --images-dir 1 -t $(pidof memcached) --shell-job --tcp-established -v0 &&
echo OK
OK
criu@criu-dev:~/dumps/memcached
$ sudo criu pre-dump --images-dir 2 --prev-images-dir ../1 --track-mem -t $(pidof memcached) --
shell-job --tcp-established -v0 && echo OK
OK
criu@criu-dev:~/dumps/memcached
$ ls -l */pages*
-rw-r--r-- 1 root root 475136 Oct 18 09:52 1/pages-1.img
-rw-r--r-- 1 root root 1642496 Oct 18 09:52 2/pages-1.img
criu@criu-dev:~/dumps/memcached
$ sudo criu pre-dump --images-dir 3 --prev-images-dir ../2 --track-mem -t $(pidof memcached) --
shell-job --tcp-established -v0 && echo OK
OK
criu@criu-dev:~/dumps/memcached
$ ls -l */pages*
-rw-r--r-- 1 root root 475136 Oct 18 09:52 1/pages-1.img
-rw-r--r-- 1 root root 1642496 Oct 18 09:52 2/pages-1.img
-rw-r--r-- 1 root root 614400 Oct 18 09:53 3/pages-1.img
criu@criu-dev:~/dumps/memcached
$ sudo criu dump --images-dir 4 --prev-images-dir ../3 --track-mem -t $(pidof memcached) --shel
l-job --tcp-established -v0 && echo OK
OK
criu@criu-dev:~/dumps/memcached
$ sudo criu lazy-pages --images-dir 4 -v0 &
[1] 24259
criu@criu-dev:~/dumps/memcached
$ sudo criu restore --shell-job --tcp-established --images-dir 4 --lazy-pages -d && echo OK
[1]+ Done sudo criu lazy-pages --images-dir 4 -v0
OK
criu@criu-dev:~/dumps/memcached
$ █
```

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

```
ailable by default. Update your scripts to load br_netfilter if you need this.
[ 6.700486] nf_conntrack version 0.5.0 (65536 buckets, 262144 max)
[ 6.770707] ip6_tables: (C) 2000-2006 Netfilter Core Team
[ 6.781687] Bits 55-60 of /proc/PID/pagemap entries are about to stop being page-shift some time soon. See the linux/Documentation/vm/pagemap.txt for details.
[ 6.784485] The pagemap bits 55-60 has changed their meaning! See the linux/Documentation/vm/pagemap.txt for details.
[ 6.835753] tun: Universal TUN/TAP device driver, 1.6
[ 6.837770] tun: (C) 1999-2004 Max Krasnyansky <maxk@qualcomm.com>
[ 6.881899] cgroup: new mount options do not match the existing superblock, will be ignored
[ 6.979365] IPv6: ADDRCONF(NETDEV_UP): veth1: link is not ready
[ 7.111814] Netfilter messages via NETLINK v0.30.
[ 7.126345] ctnetlink v0.93: registering with nfnetlink.
[ 7.145056] br0: port 1(veth0) entered blocking state
[ 7.146104] br0: port 1(veth0) entered disabled state
[ 7.147289] device veth0 entered promiscuous mode
[ 7.148511] br0: port 1(veth0) entered blocking state
[ 7.149597] br0: port 1(veth0) entered forwarding state
[ 7.158055] IPv6: ADDRCONF(NETDEV_CHANGE): veth1: link becomes ready

0 adrian@rhlx01:~
Wed Oct 10 08:44:16 2018 OPTIONS IMPORT: route options modified
Wed Oct 10 08:44:16 2018 OPTIONS IMPORT: peer-id set
Wed Oct 10 08:44:16 2018 OPTIONS IMPORT: adjusting link_mtu to 1624
Wed Oct 10 08:44:16 2018 OPTIONS IMPORT: data channel crypto options modified
Wed Oct 10 08:44:16 2018 Data Channel: using negotiated cipher 'AES-256-GCM'
Wed Oct 10 08:44:16 2018 Outgoing Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
Wed Oct 10 08:44:16 2018 Incoming Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
Wed Oct 10 08:44:16 2018 ROUTE_GATEWAY 192.168.122.1/255.255.255.0 IFACE=eth0 HWADDR=52:54:00:35:0b:f1
Wed Oct 10 08:44:16 2018 TUN/TAP device tun0 opened
Wed Oct 10 08:44:16 2018 TUN/TAP TX queue length set to 100
Wed Oct 10 08:44:16 2018 do_ifconfig, tt->did_ifconfig_ipv6_setup=0
Wed Oct 10 08:44:16 2018 /sbin/ip link set dev tun0 up mtu 1500
Wed Oct 10 08:44:16 2018 /sbin/ip addr add dev tun0 local 172.31.0.6 peer 172.31.0.5
Wed Oct 10 08:44:16 2018 /sbin/ip route add 172.31.0.0/24 via 172.31.0.5
Wed Oct 10 08:44:16 2018 WARNING: this configuration may cache passwords in memory -- use the auth-nocache option to prevent this
Wed Oct 10 08:44:16 2018 Initialization Sequence Completed

1 adrian@dcbz:~
(00.326304) cg: ^- Dumping cpuacct,cpu of /
(00.326310) cg: ^- Dumping cpuset of /
(00.326316) cg: ^- Dumping devices of /
(00.326321) cg: ^- Dumping freezer of /
(00.326326) cg: ^- Dumping hugetlb of /
(00.326332) cg: ^- Dumping memory of /
(00.326337) cg: ^- Dumping name=systemd of /system.slice/rc-local.service
(00.326343) cg: ^- Dumping net_prio,net_cls of /
(00.326349) cg: ^- Dumping perf_event of /
(00.326354) cg: ^- Dumping pids of /
(00.326402) cg: Writing CG image
(00.326596) sk unix: Dumping external sockets
(00.326610) Writing image inventory (version 1)
(00.326865) Running post-dump scripts
(00.326874) Unfreezing tasks into 2
(00.326880) Unseizing 1728 into 2
(00.336268) Writing stats
(00.336451) Dumping finished successfully
Connection to server01 closed by remote host.
Connection to server01 closed.
adrian@dcbz$
```


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!