Container live-migration unsolved issues

Andrei Vagin <avagin@virtuozzo.com>

Agenda

- Last year for CRIU
- What is wrong with pre-dump
- The P.Haul golang library
- Live-migration of Docker containers

Last year for CRIU

- CRIU for mainframes (s390)
- Support for x32 compatible mode
- Nested namespaces
- Lazy restore (userfaultfd)
- Image cache, image proxy
- Socket SCM
- TCP transitional states

pre-dump: how it works now

- Freeze all processes
- Splice their memory into pipes
- Reset soft-dirty memory tracker
- Resume processes
- Dump memory from pipes

pre-dump: pros and cons

Pros

- Freeze processes for a short time

Cons

- A size of each pipe is limited
- A number of pipes are limited
- Memory in pipes are locked
- Need to inject a parasite code

pre-dump: process_vm_readv()

Pros

- No need to inject a parasite code

Cons

- An extra copy of data into a user-space buffer

New syscall: process_vmsplice

- ```
  ssize_t process_vmsplice(pid_t pid, int fd,
                           const struct iovec *iov,
                           unsigned long nr_segs,
                           unsigned int flags)
  ```
- *a hybrid of process_vm_readv() and vmsplice()*
- No need to inject a parasite code
- Can dump memory iteratively
  - - small per-iteration overhead
  ```

P.Haul golang library

- https://github.com/xemul/criu/tree/criu-dev/phaul
- Can be integrated into projects (Docker, K8s)
- Can use existing infrastructure
  - create channels between nodes
  - handle volumes and images
  - ...

Live-migration of Docker containers

- **Images**
- The destination node has to have the same image
- **Volumes**
- Local volumes have to be transferred to a destination node
- Shared volumes have to be mounted on destination node. This can be solved with snapshots.
- **Network**
- Need to restore with the same IP address

Thank You!