Google

Update on Task Migration at Google

Linux Plumbers Conference 2019

Kamil Yurtsever (kyurtsever@google.com)
and Michał Cłapiński (mclapinski@google.com)

Background

Google started using CRIU for migration of workloads in Borg in 2018.

Presented at Linux Plumbers Conference 2018 in Vancouver.

We have expanded the scope and improved the technology over the last year.

This presentation will go over the changes we have made and our requests for improvement from the kernel side.

Assumptions

CRIU not running as root.

Migrating process tree within the namespaces.

Migration finishing within minutes.

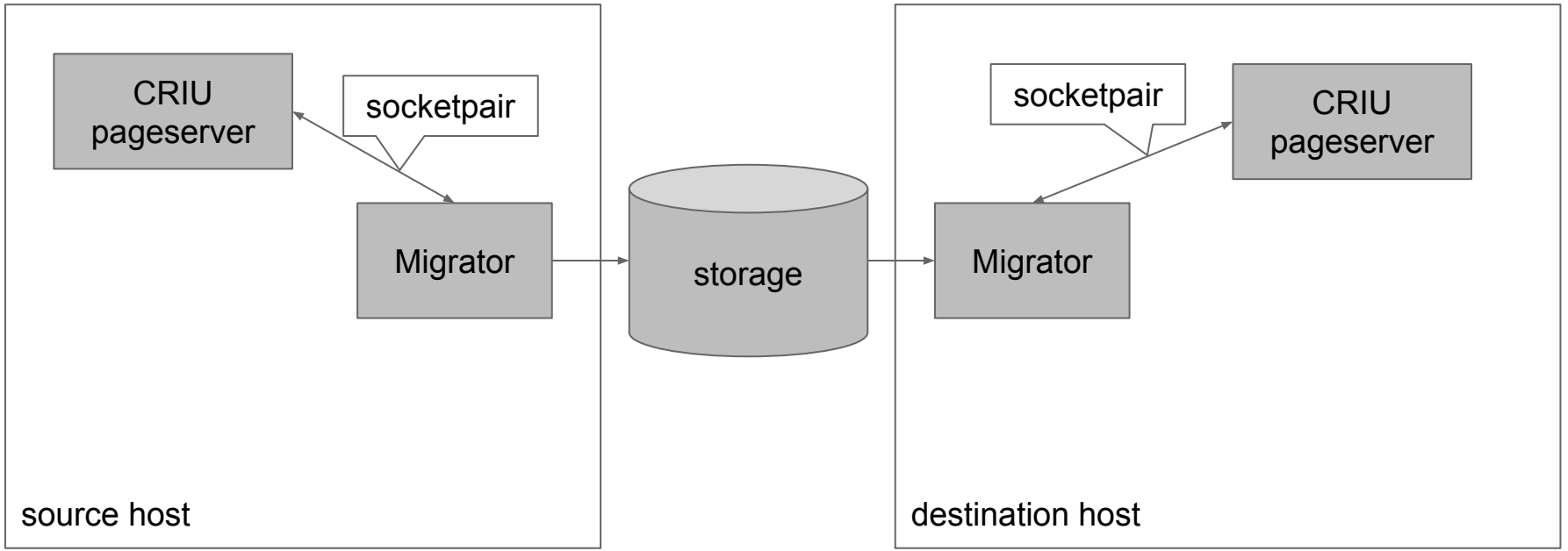
What’s new

Streaming migration

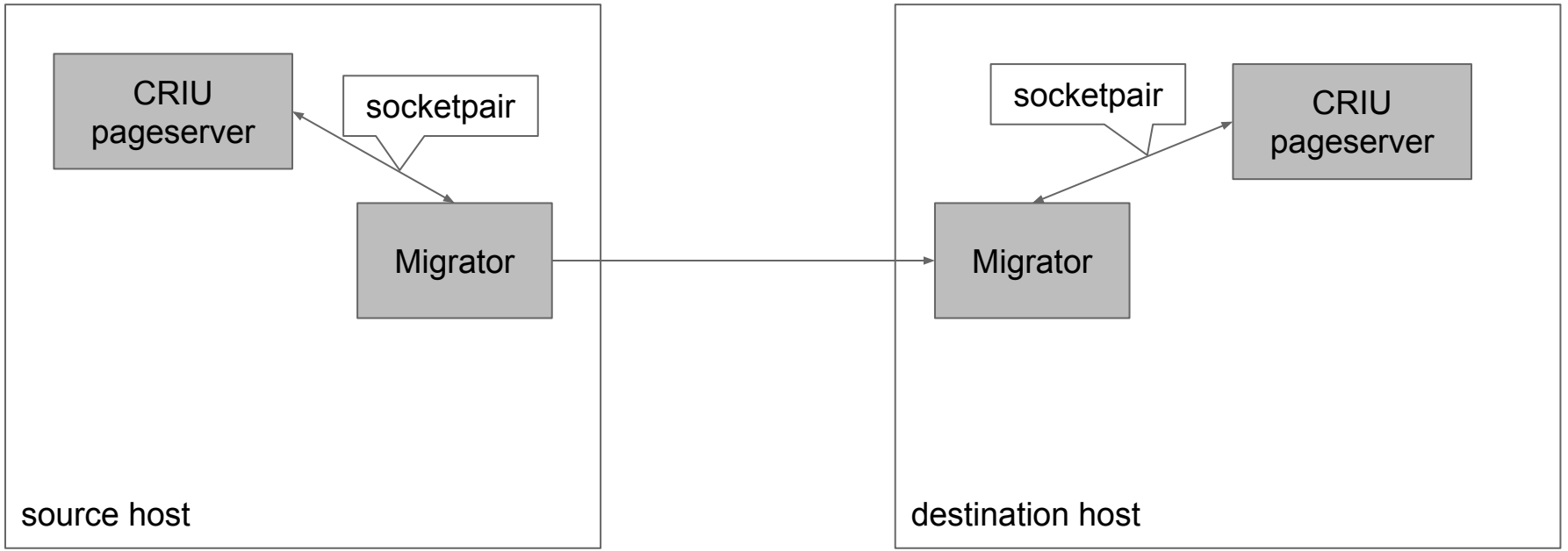
Mostly replaced the migration through storage.

Relies on CRIU pageserver with added encapsulation layer.

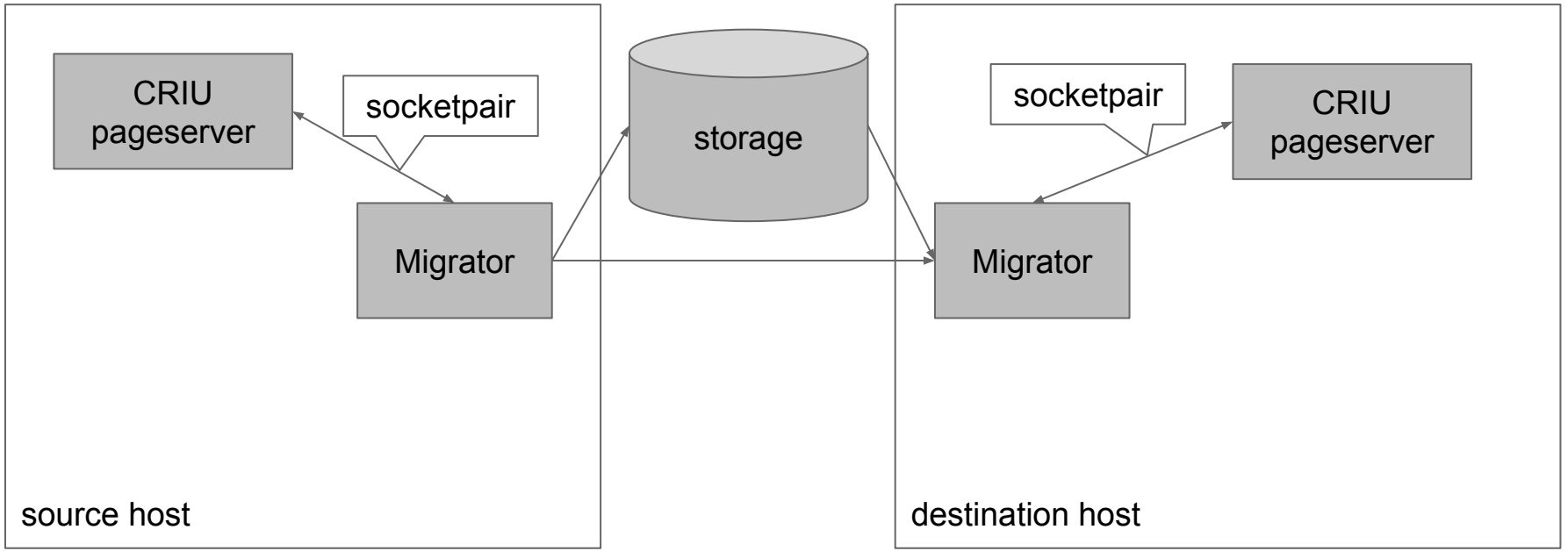
Migration through storage



Direct streaming migration



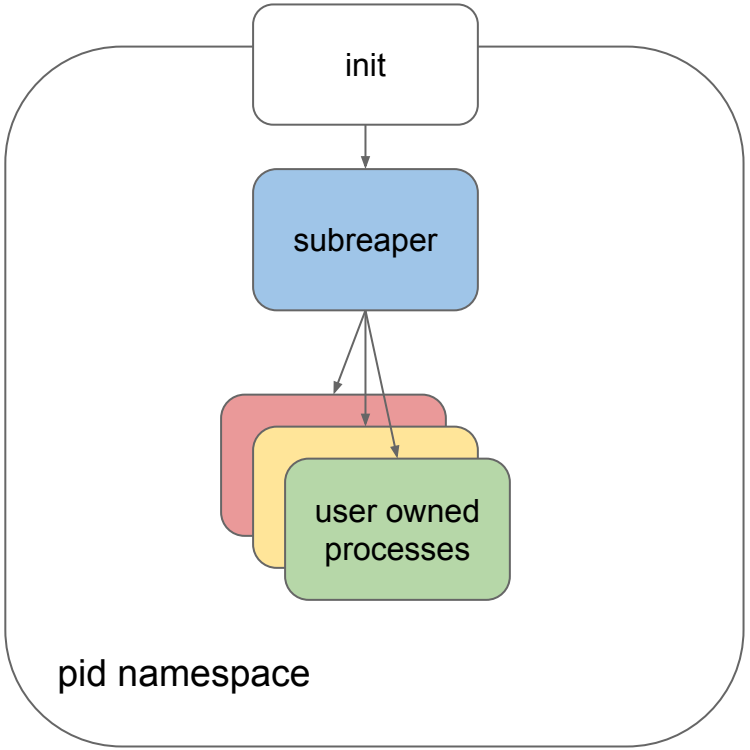
Flexibility to flip between the versions



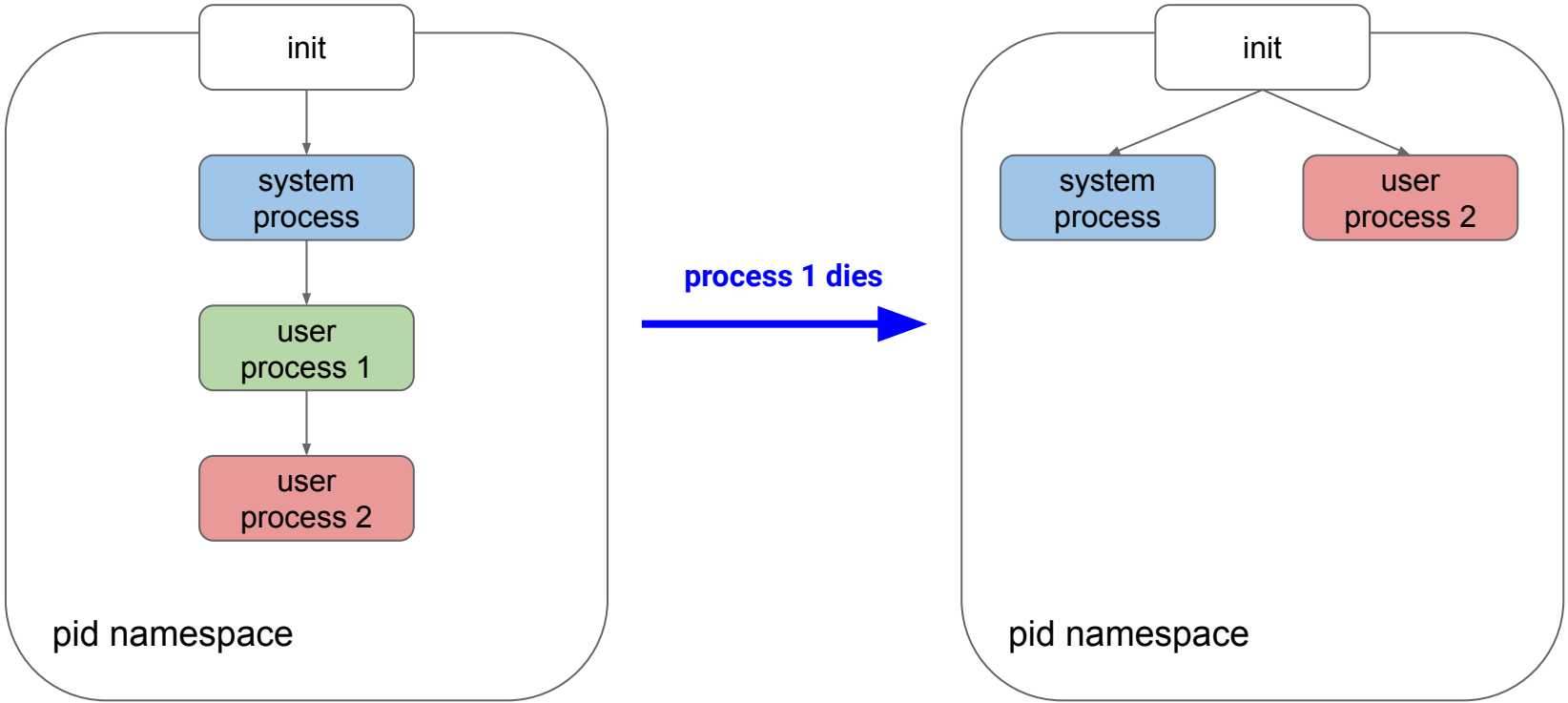
Google

Subreaper support

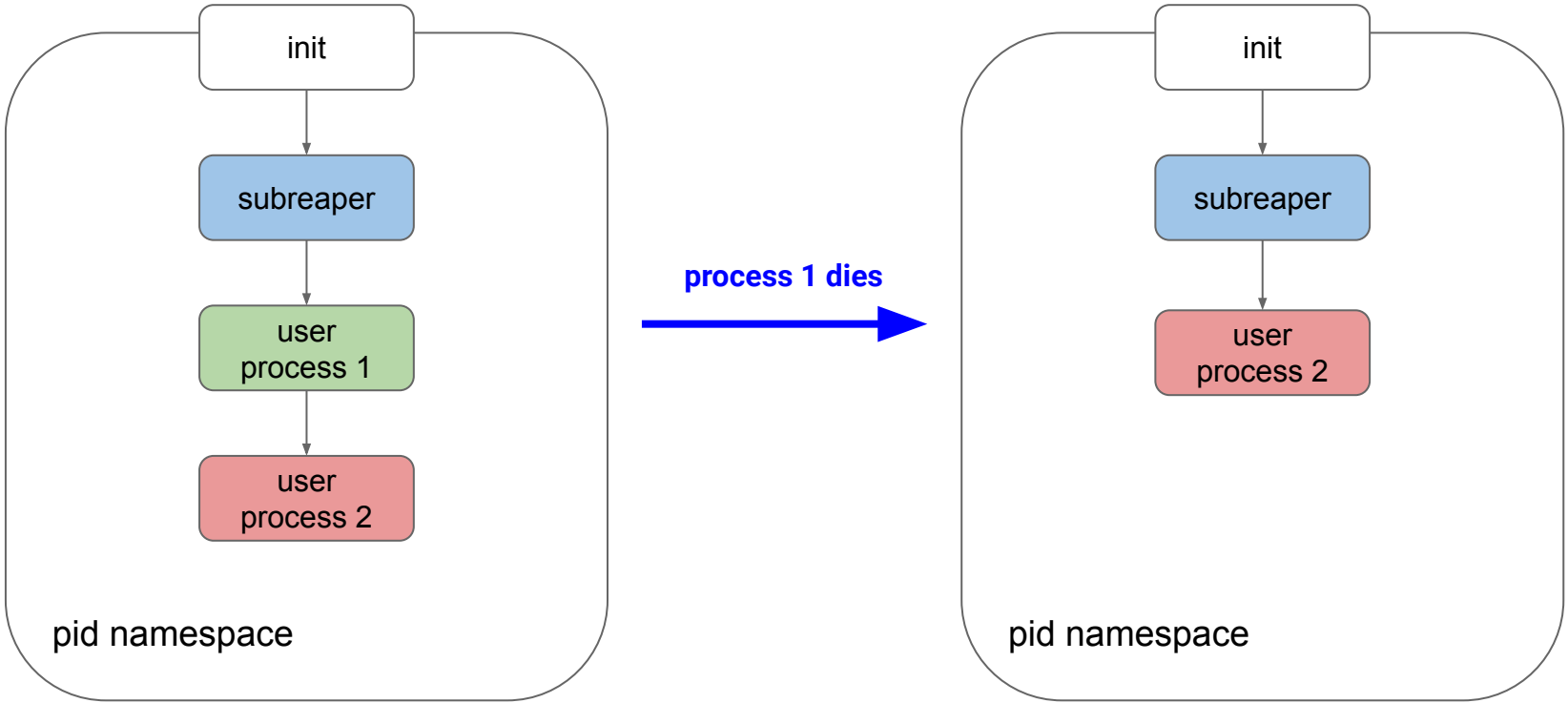
PR_SET_CHILD_SUBREAPER is especially useful if the migration doesn't entail namespaces.



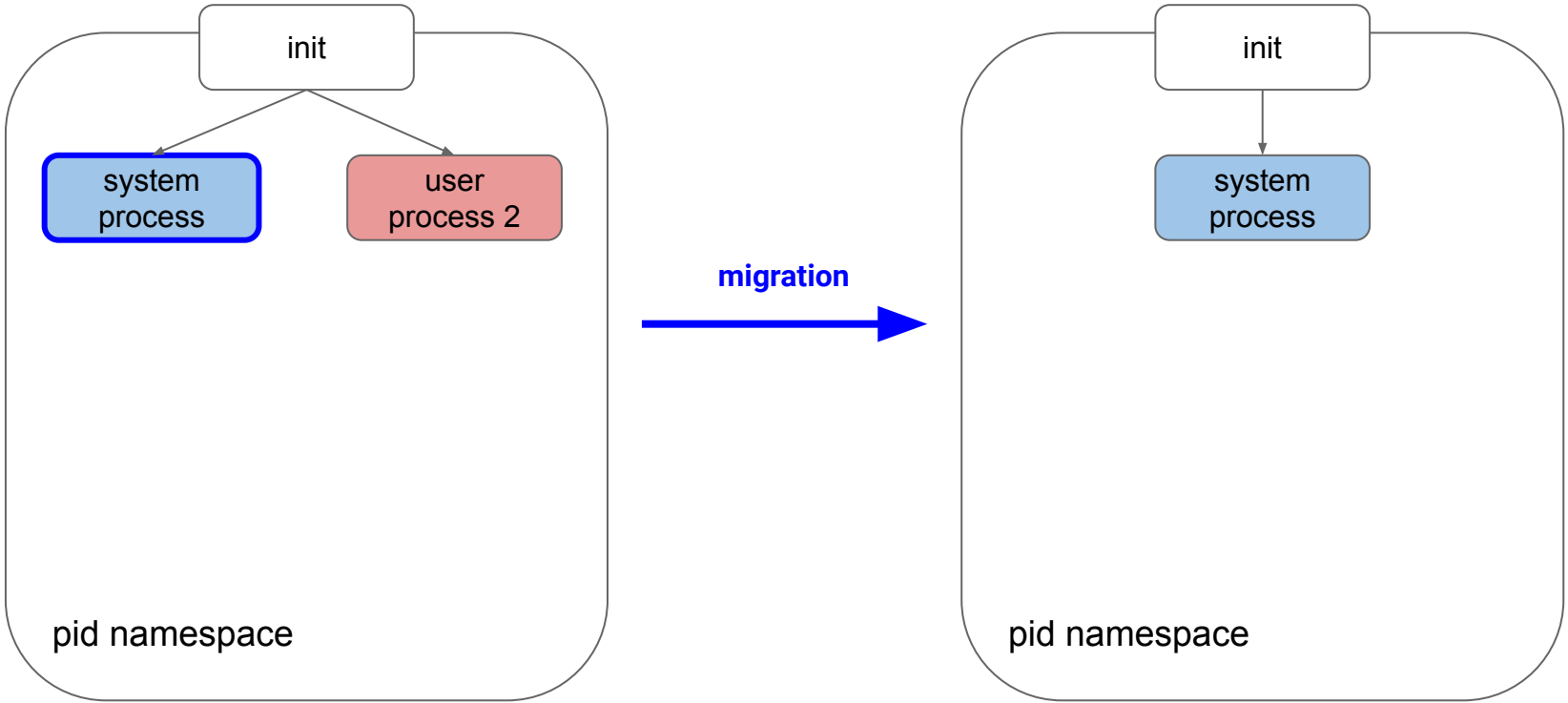
Reparenting without a subreaper



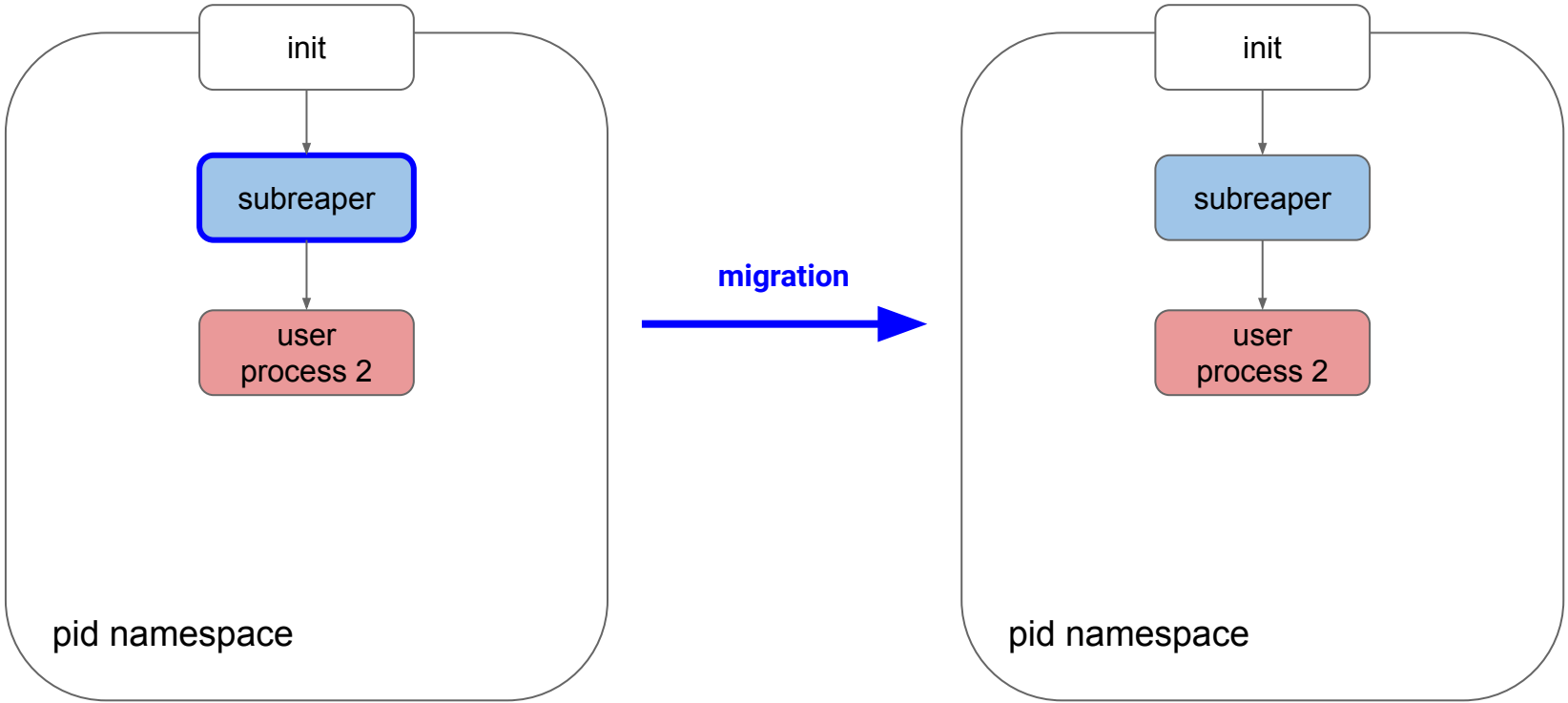
Reparenting with a subreaper



Migration without a subreaper



Migration with a subreaper



Subreaper support

Tricky interaction with CRIU restoring order.

Two possibilities for restoring this attribute:

1. Before forking - can’t because that would break migration of orphaned processes
2. After forking - can’t because of a bug in old kernels

Thanks to Pavel Tikhomirov

cgroup migration with limited privileges

cgroups can be delegated to users which allows checkpoint and restore without root privileges.

However CRIU needs a clean view of cgroup mounts.

We’ve added a way to provide a clean view to CRIU externally.

Migration with IP change

We don’t migrate the IP address together with the container.

Nevertheless our users handle that well using appropriate library code.

We are happy that CRIU keeps working even if the IP is changed.

We would like for this use case to be supported as it is today.

CRIU error messaging

Errors don’t provide a clear indication of what area has failed.

We need to parse the whole log to recover the kind of failure.

Categorization of errors and a clear location for the final message.

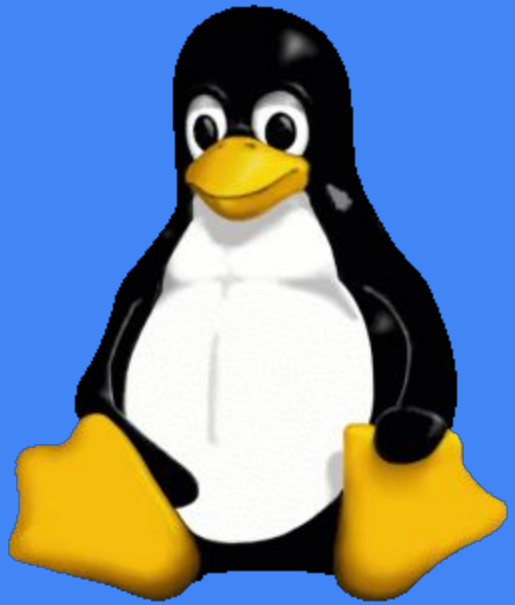
We care about failures - every bit of debugging info is useful.

O_PATH support

CRIU itself often uses O_PATH file descriptors.

However it doesn’t migrate O_PATH file descriptors correctly.

Patch coming soon.

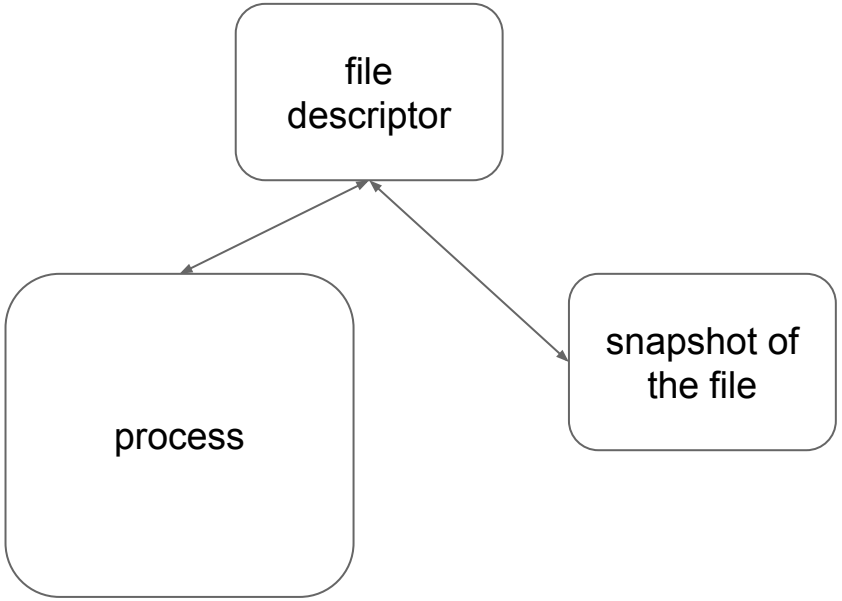


Migrating the state of files from virtual filesystems

Migrating regular files is simple.

Virtual files have stable state kept on open but no standard way to recover that data.

More importantly no way to restore that data.



CAP_RESTORE

Some interfaces are heavily restricted.

- /proc/PID/map_files (CAP_SYS_ADMIN)
- /proc/sys/kernel/ns_last_pid (CAP_SYS_ADMIN)

We would be happy to open these APIs under CAP_RESTORE.

Time Namespace

Isolating time would be great.

We workaround it in libraries.

Isolating the TSC register is desired too.

Write only APIs: rseq, cgroup v1 event API

rseq is a new syscall.

cgroup v1 is an old API

Both don’t give a way to discover the kernel state.

Can’t migrate without user/library help.

Long term maintainability

How can we make migration easier to maintain?

Can we have some APIs to dump and restore opaque kernel data?

Is there something that we can do to make sure that new features are built with migration in mind?

Q&A

Thank you!

Image Credits:
https://www.nps.gov/subjects/fishing/images/16333875678_f85416a0b9_o-1.jpg?maxwidth=1200&maxheight=1200&autorotate=false
https://storage.needpix.com/rsynced_images/columns-656322_1280.jpg
https://upload.wikimedia.org/wikipedia/commons/thumb/8/83/Winding_path_%2814354572446%29.jpg/1200px-Winding_path_%2814354572446%29.jpg
https://cdn.pixabay.com/photo/2017/02/12/21/29/false-2061132_960_720.png
https://cdn.pixabay.com/photo/2018/04/21/20/46/time-3339479_960_720.jpg
https://cdn.pixabay.com/photo/2015/11/03/08/59/search-1019904_960_720.jpg
https://cdn.pixabay.com/photo/2016/12/18/12/49/cyber-security-1915628_960_720.png
https://upload.wikimedia.org/wikipedia/commons/thumb/4/4e/Flat_restart_icon.svg/512px-Flat_restart_icon.svg.png
https://upload.wikimedia.org/wikipedia/commons/f/fd/Singapore_road_sign_-_Informatory_-_One_way_street_to_the_left.svg