

Emilua 0.7 released

Most of the code for this new release were requirements identified while working on [packxd](#). Emilua should be in a pretty good shape to spawn Linux containers now.

Documentation pages that were too bloated were also split into smaller pages (e.g. sandboxing internals, `init.script`), and a new tutorial on filesystem was added.

There's also a new [plugin to create seccomp programs](#) (useful to further limit what containers can do).

Added

- Add seccomp support.
- Add `filesystem.mkdir()` to complement `filesystem.create_directory()`.
- `filesystem.mode()` accepts new arguments now.
- Add `filesystem.chroot()`.
- `filesystem.current_working_directory()` accepts `file_descriptor` objects on UNIX now.
- Add extra optional parameter to `filesystem.mknod()`.
- Add `filesystem.clock.epoch()`. It's useful to set the last modification date of every file in some directory for the purposes of a reproducible build or something. However there are more attributes besides last-write-time you need to care about if you're planning to play with reproducible builds (be warned!).
- Add `filesystem.clock.unix_epoch()` and `filesystem.clock.now()`.
- Add more POSIX bindings to `init.script` API.
- Add the `flock()` family to `file.stream` and `file.random_access`.
- Now it's possible to configure Landlock mode for the calling process or `system.spawn()` subprocesses.
- Add `byte_span` methods for primitive types serialization (e.g. reading i32le from a 4-sized buffer). It also works as an endianness handling interface. 64-bit integers are omitted from the interface because LuaJIT only offers a hacky way to handle them.

Changed

- Make `subprocess.pid` nullable. That's useful for synchronization when multiple fibers are observing parts of subprocess state.
- Allow `file_descriptor.close()` to be called multiple times in a row.
- Change `filesystem.copy_file()` parameters.
- Change every name in the module `filesystem` from `hard_*` to `hard*` (e.g. `create_hard_link()` to

`create_hardlink()`). This C++17 convention is dumb and Python's `pathlib` is the one who got it right.

- Change default `record_separator` in `stream.scanner` to `"\n"`.
- Always start subprocess-based actors with `umask 022`.
- Change `system.spawn()` parameters from `nsenter_*` to `setns_*`.

Fixed

- Close file descriptors from builtin PID1 so EPIPE propagates sooner.
- Fix races in `filesystem.current_working_directory()`. Now `fchdir()` is used.
- Small documentation issues.
- Avoid potential IO double-flush on FreeBSD after `fork()`.