Morello Linux

Technical Update

Vincenzo Frascino
08/11/2023

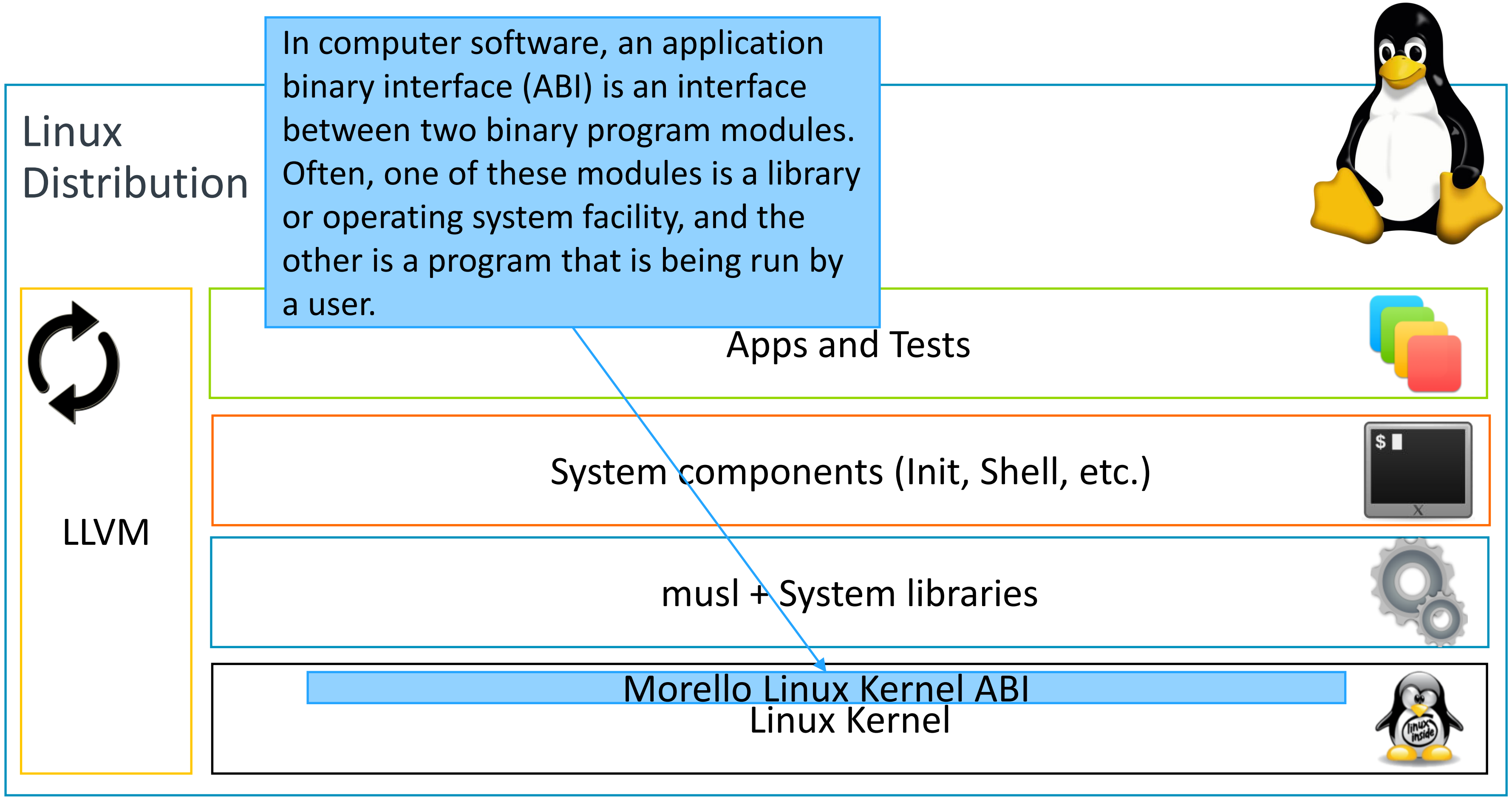
Morello Linux - Summary

- What have we achieved?
- Pure Capability user ABI Technical Update
- Status of Userspace
- What’s next? - Future opportunities for the Ecosystem Research
- Build a Morello Community

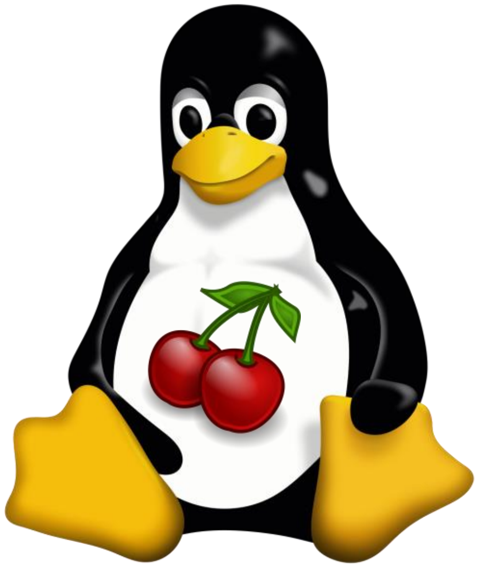
Morello Linux

What have we achieved?

Morello Linux – PureCap Application Binary Interface (PCuABI)



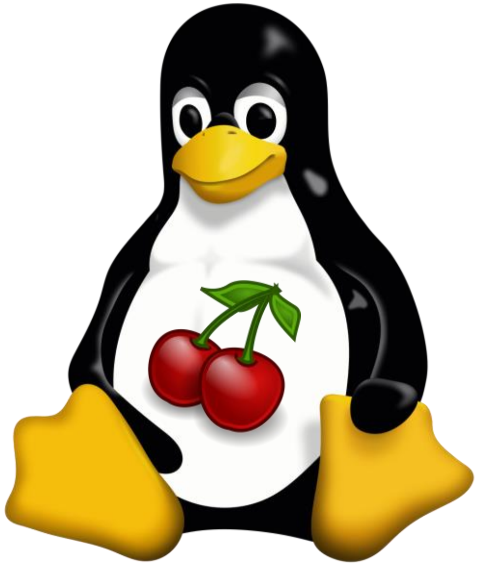
In computer software, an application binary interface (ABI) is an interface between two binary program modules. Often, one of these modules is a library or operating system facility, and the other is a program that is being run by a user.



Morello Linux – *"Our Motto"*

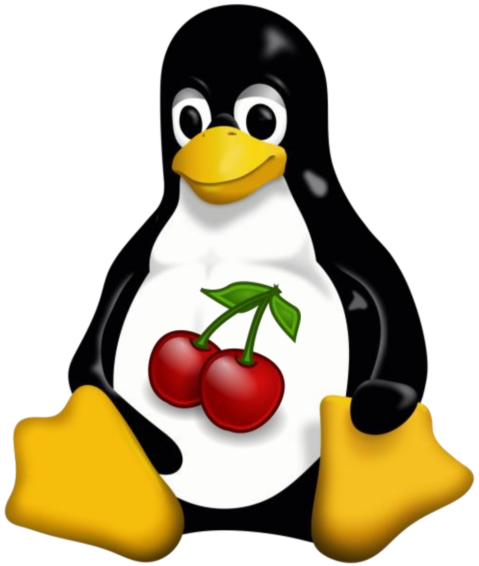
Let Linux developers focus on their capability based usecase.

Meaning: Enable Linux developers to develop/port their Linux applications on the Morello architecture and researchers to target security and performance investigations in userspace.



Morello Linux – How do we want to achieve that?

- Design and implement a future proof ABI with Capability Support.
- Provide Linux support for the porting of userspace applications.
- Help the Linux community to standardize and enable 128-bit architectures.

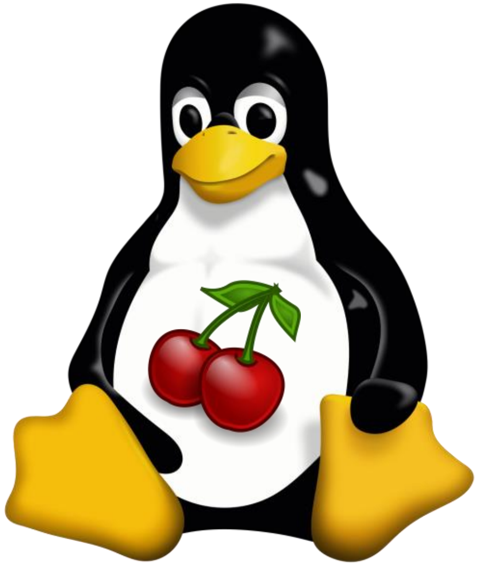


Morello Linux – Enable Developers

Define a Morello pure-capability ABI, or PCuABI: all pointers at the kernel-user interface are 129-bit capabilities, instead of 64-bit integers.

Fundamental design goals for the new ABI:

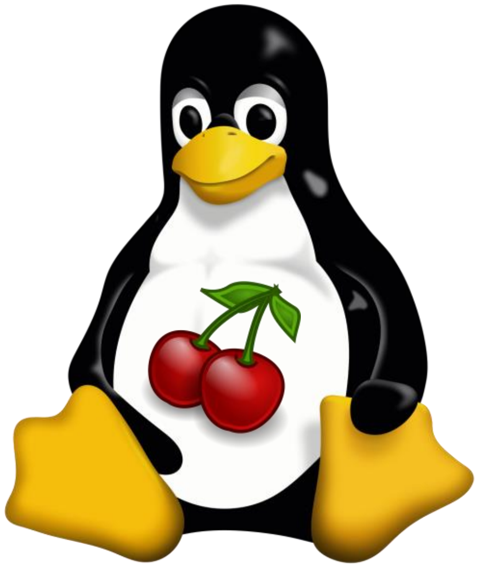
- To provide Native support for UserSpace applications built in the pure-cap ABI.
- To improve memory safety at the kernel-user boundary, by leveraging the properties of capabilities, and to ensure that the capability model cannot be weakened through the kernel interface.
- Converge with CheriBSD's pure-cap ABI.



Morello Linux – Create a community

Challenges:

- Define a lightweight process accepted by the existing opensource community.
- Simplify the submission of new changes and make it familiar to developers.
- Identify a common place to store the information, to make it easily accessible.
- Explore ways to make everyone feel part of the same Morello community (no differences in between internal and external developers).

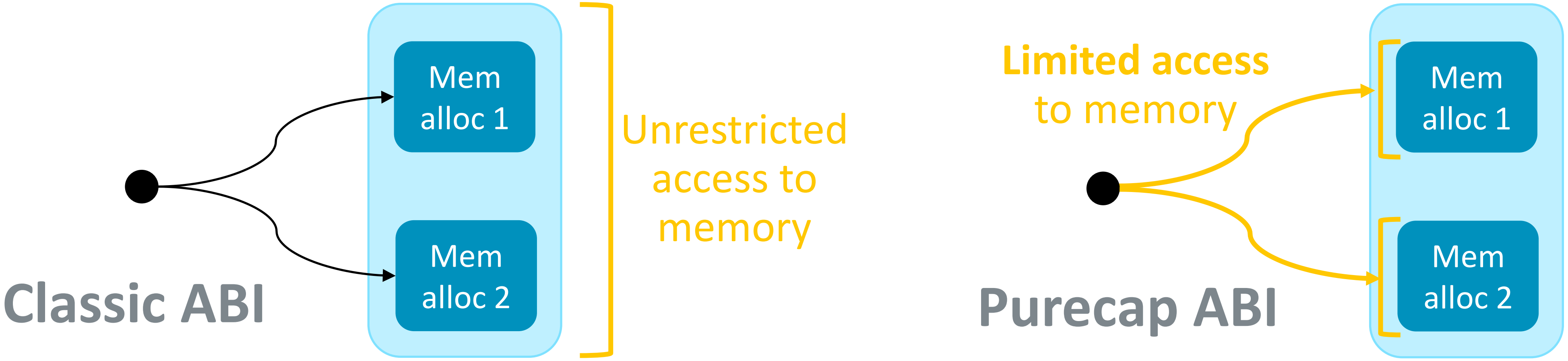


Morello Linux

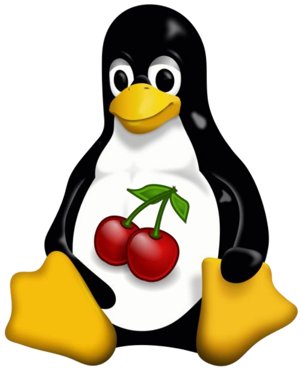
Pure Capability user ABI Technical Update

The pure-capability model

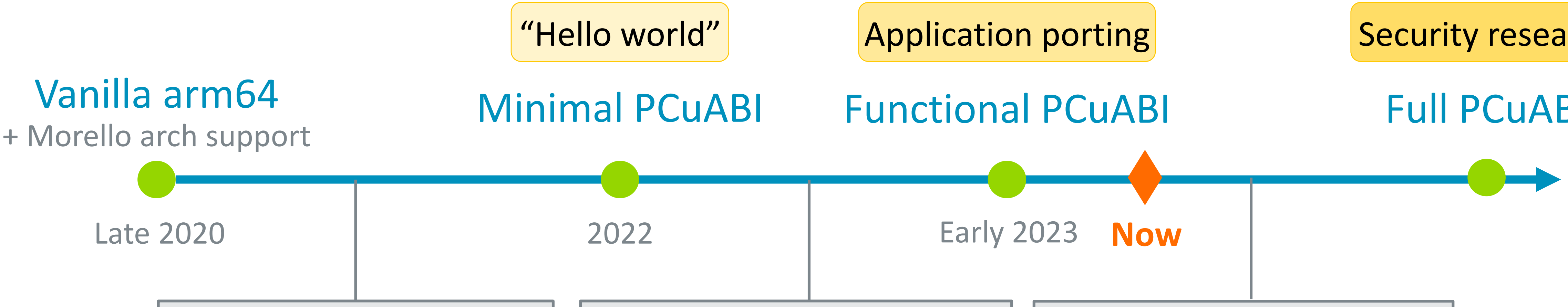
- New pure-capability (purecap) ABI: all pointers are implemented as capabilities



- Easiest way to reap the benefits of the CHERI model
  - Provides comprehensive spatial memory safety
  - Transparent switch for most application code
- Requires support throughout the system
  - Completely new ABI: think 32-bit ⟶ 64-bit transition!



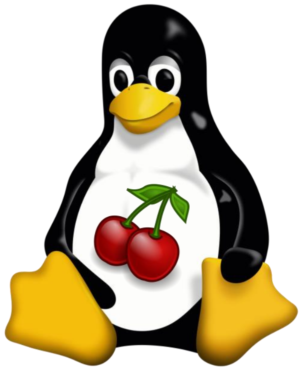
Hybrid kernel – implementation status



- Representation of user pointers as caps
- Core PCuABI plumbing
  - Syscall dispatch
  - ELF loader
  - Signal handling
- Adaptation of essential syscalls
- 64-bit COMPAT

- Expansion of PCuABI-compatible syscalls
  - Ensure propagation of caps
  - Adapt some uAPI structs
- Bug-hunting
  - LTP, Musl tests
  - Both PCuABI and compat64

- Enforcement of capability model
  - Authorise all accesses to user memory through a cap
  - Minimise bounds and permissions of provided caps
- Address space management through caps
- …

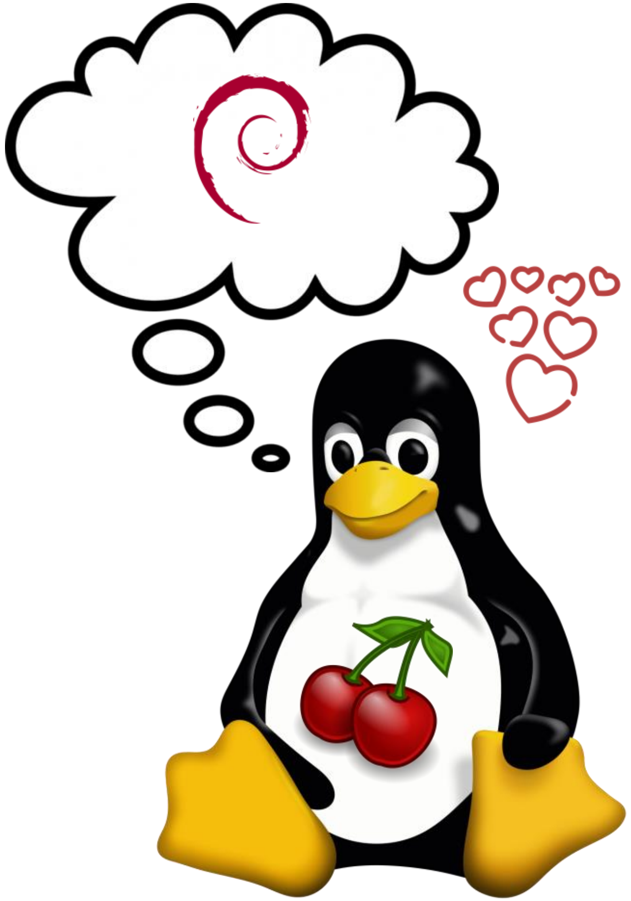


Morello Linux

Status of Userspace

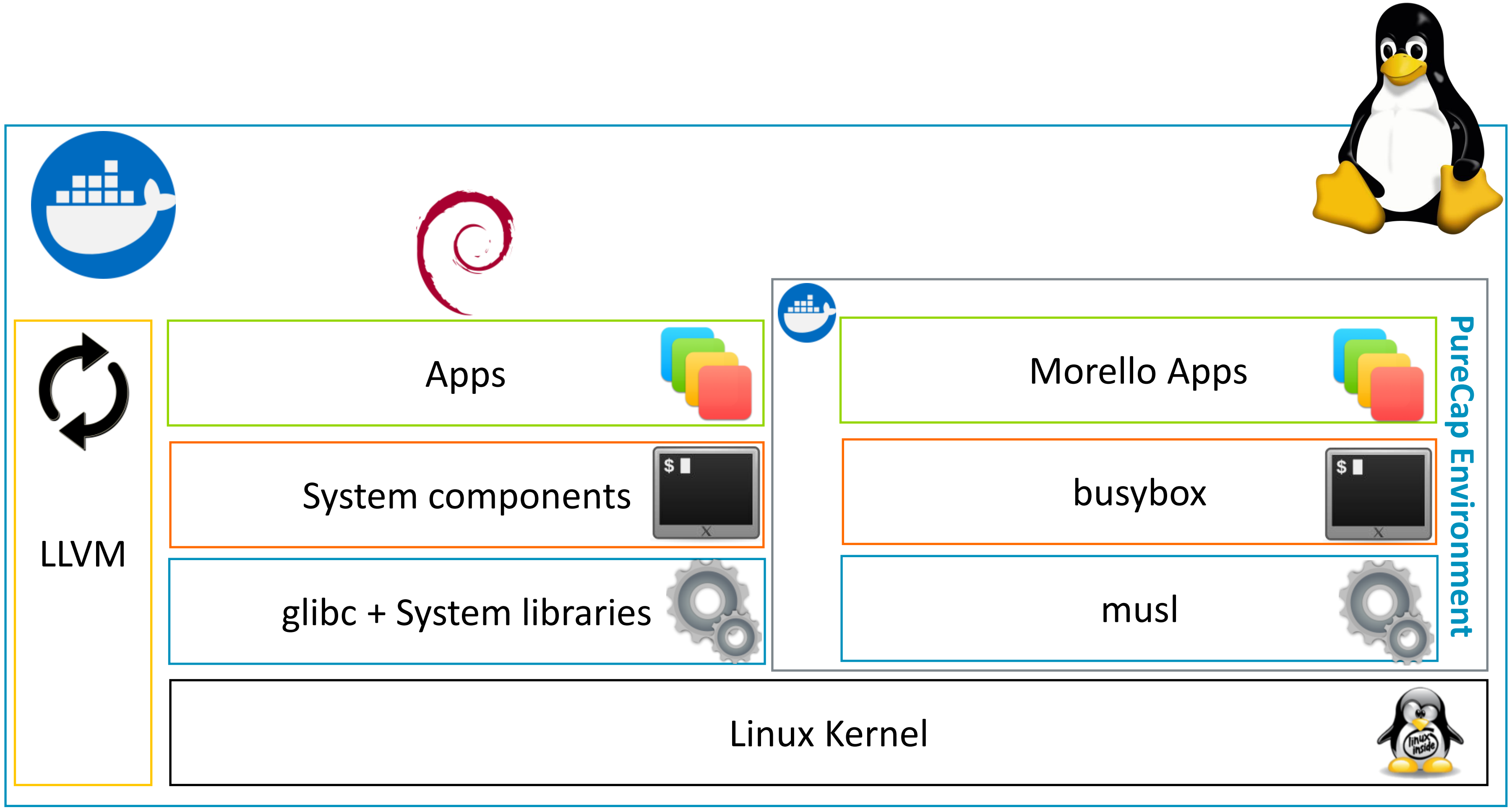
Morello Linux

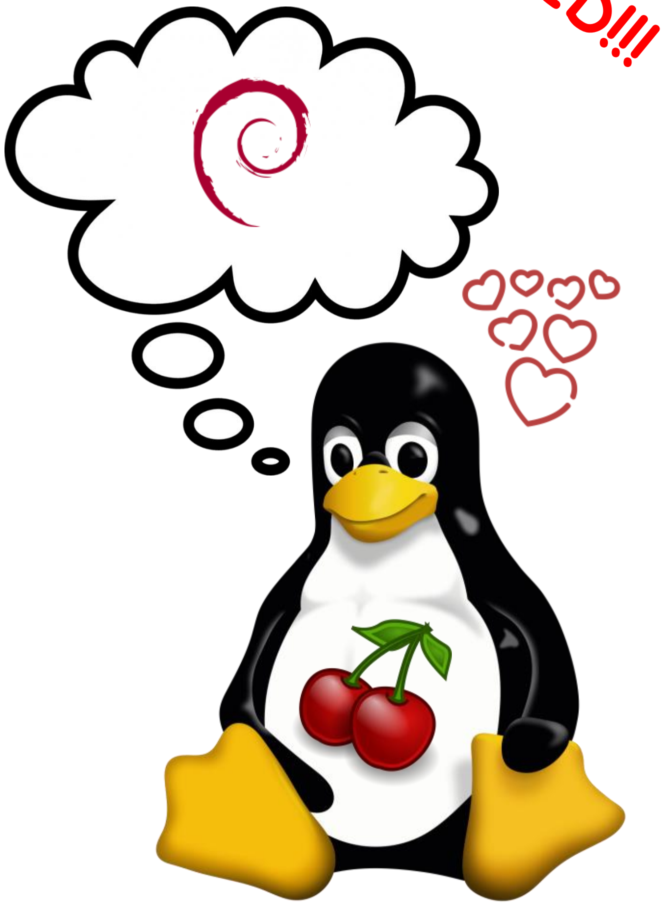
HELP WANTED!!!



Morello Linux – Hybrid Software Stack

HELP WANTED!!!





Morello Linux – Initial Release in April 2023

HELP WANTED!!!

Morello SDK

In less than 10 minutes you should be able to setup a docker container with everything you need to build an application for Morello.

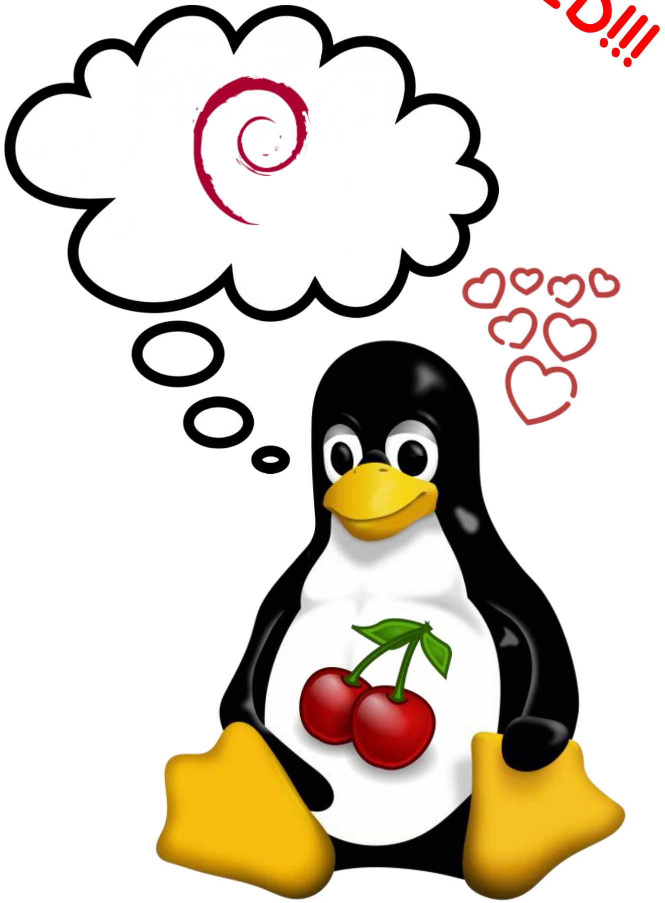
- Documentation: https://sdk.morello-project.org/
- Code repository: https://git.morello-project.org/morello/morello-sdk

Morello Linux

In less than 10 minutes you should be able to setup a docker container with everything you need to build and boot into a Morello Debian environment.

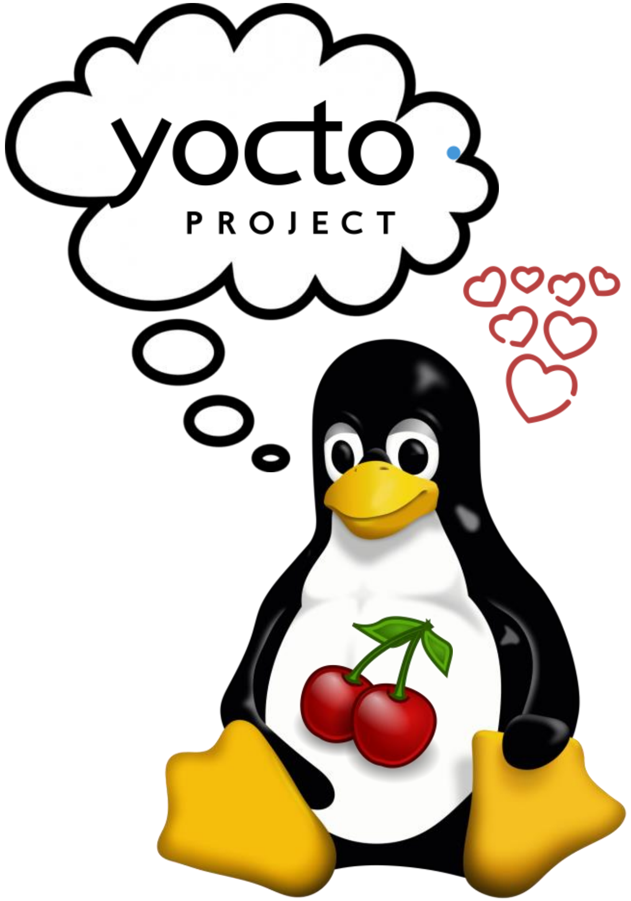
- Documentation: https://linux.morello-project.org/
- Code repository: https://git.morello-project.org/morello/morello-linux

Note: The documentation covers the instructions for Linux but if you know what you are doing and are familiar with docker no one stops you from running our solution on Windows or Mac.



Yocto Project and meta-morello

HELP WANTED!!!



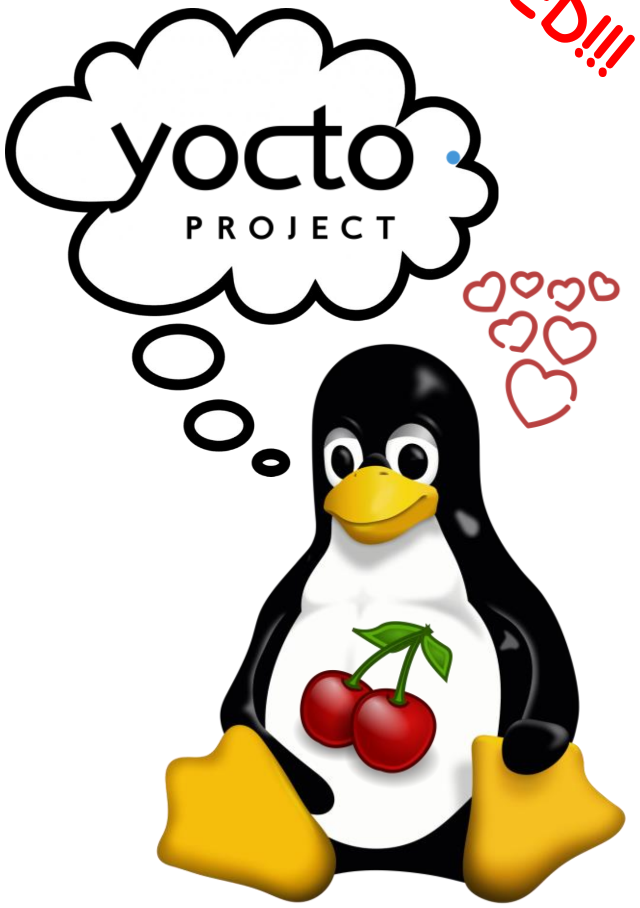
meta-morello: Making embedded systems fun again!

HELP WANTED!!!

- Morello is a capability based architecture (pointers are 129 bit).
- We are designing a Pure Capability user ABI (PCuABI) to enable user space.
- meta-morello is a community based effort to support Yocto on Morello.
- meta-morello is compliant with the PCuABI as it provides:
  - The layer required to build the firmware.
  - A Morello enabled Linux kernel.
  - A set of recipes to generate a musl/glibc based rootfs.
- meta-morello is kas based to improve the efficiency in managing dependencies.

meta-morello repo

meta-morello mailing list



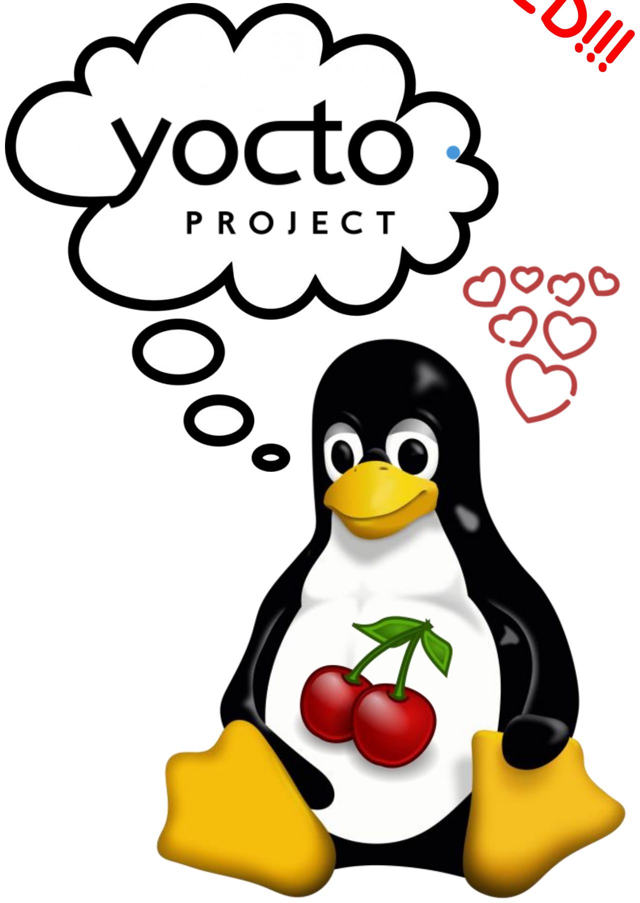
meta-morello: Supported Machines

HELP WANTED!!!

- meta-morello supports morello native applications and libraries that co-exist with aarch64 ones.
- The supported machines are:
  - morello-bsp: generates the firmware for the morello boards.
  - morello-linux-musl: generates the linux image and the rootfs with musl for morello as a libc.
  - morello-linux-glibc: generates the linux image with a generic Yocto image as the rootfs.
- The system supports 2 C libraries:
  - musl for capability aware applications
  - glibc for all of the rest

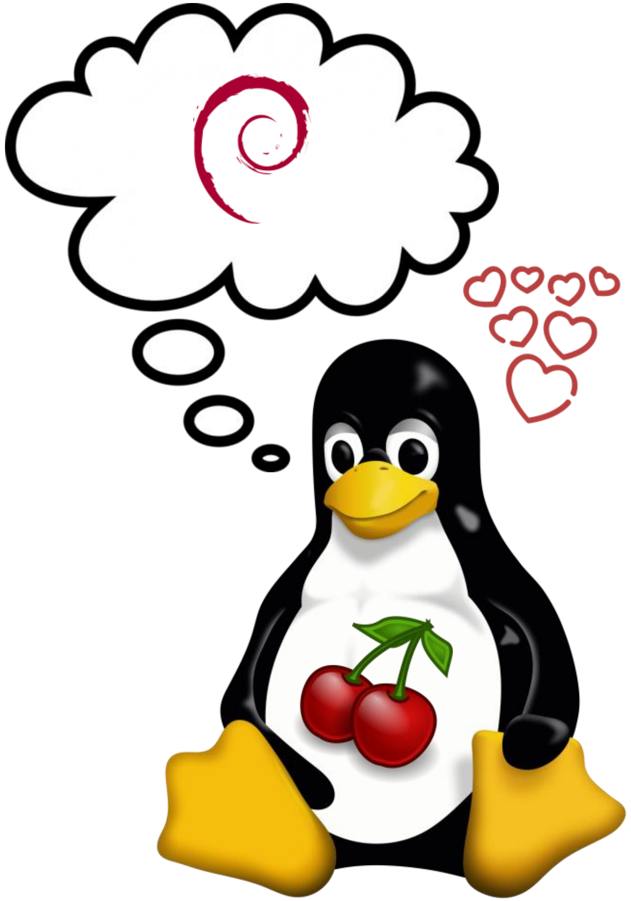
meta-morello repo

meta-morello mailing list



Morello Linux - Demo

HELP WANTED!!!



Morello Linux

What’s Next?

Opportunities for Ecosystem Research

Morello Linux – Future opportunities for the Ecosystem Research

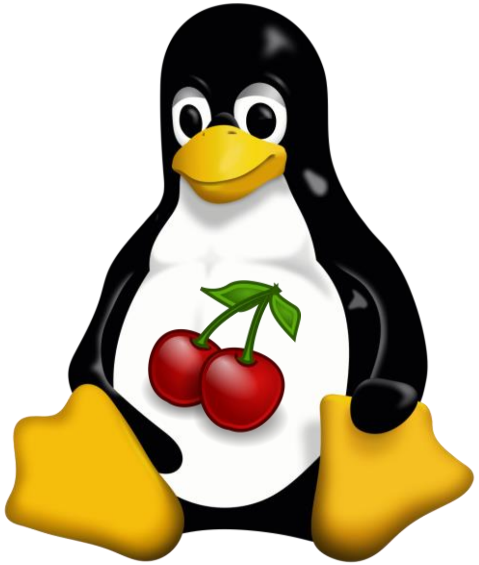
HELP WANTED!!!

Context Switches are costly: are there areas where we can leverage Morello's Security Capabilities to avoid some of them?

We identified four areas that seem worth investigation (to begin with):

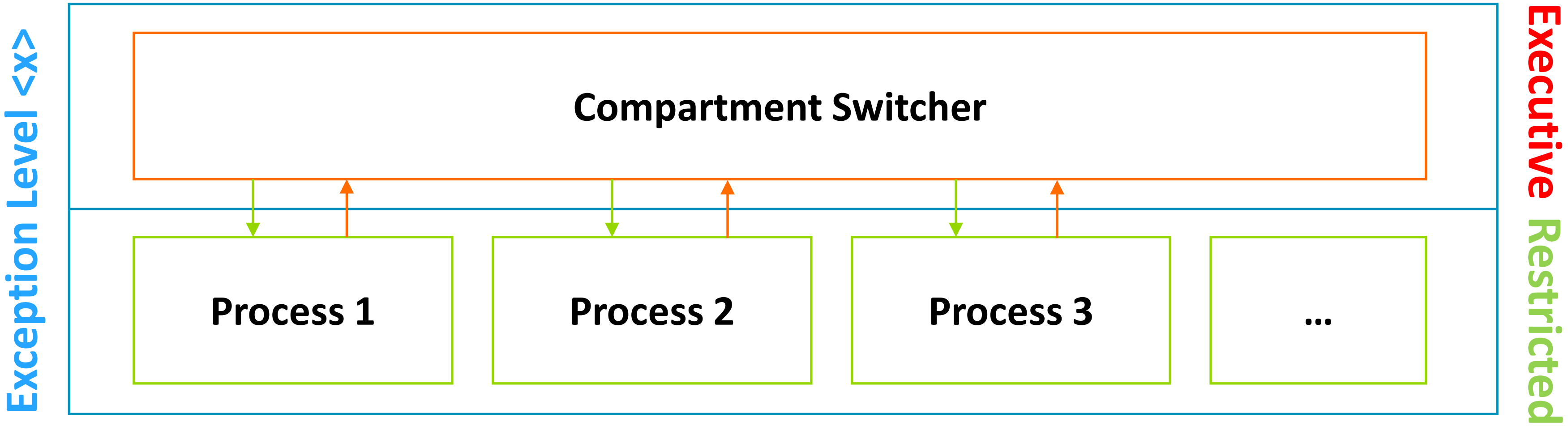
- Co-processes
- Lightweight VMs
- IO_uring
- eBPF JIT engine

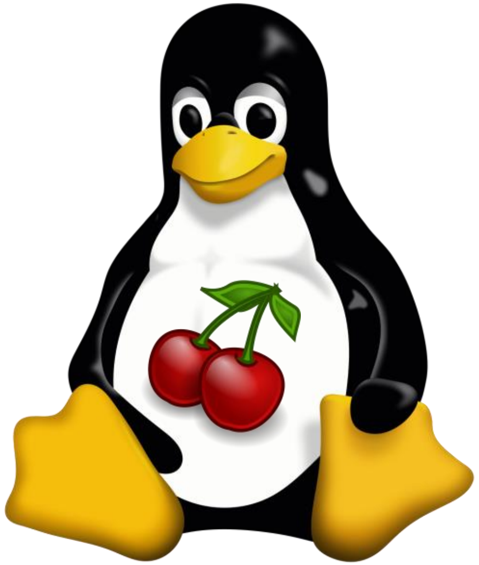
Note: The purpose of our activities is to identify opportunities for future ecosystem research. We will do our part, but we are well aware that without the help coming from the community we will not be able to deliver on these areas.



Morello Linux – Co-Processes (1/2)

HELP WANTED!!!





Morello Linux – Co-Processes (2/2)

HELP WANTED!!!

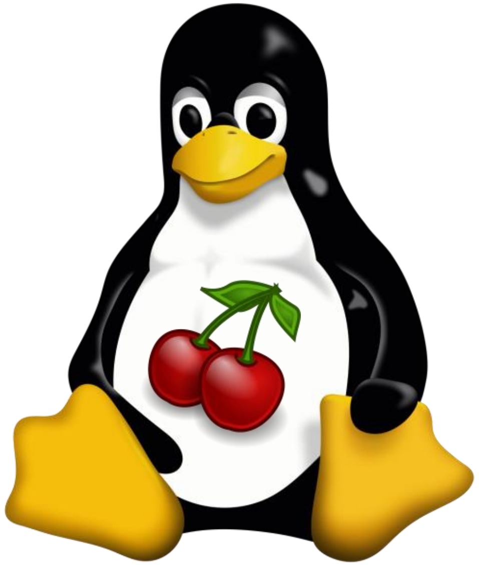
Isolate multiple processes within a single virtual address space:

- Kernel (something similar to vDSO library) or User (through a dynamically loaded library that runs in executive mode) trusted compartment switcher runs in userspace which includes:
  - inter-process memory sharing
  - compartment switcher
- The idea is that inter-process context switches take no architectural exceptions and do not need to enter the kernel.

Looks promising, due to Cambridge University's early experiments on CheriBSD.

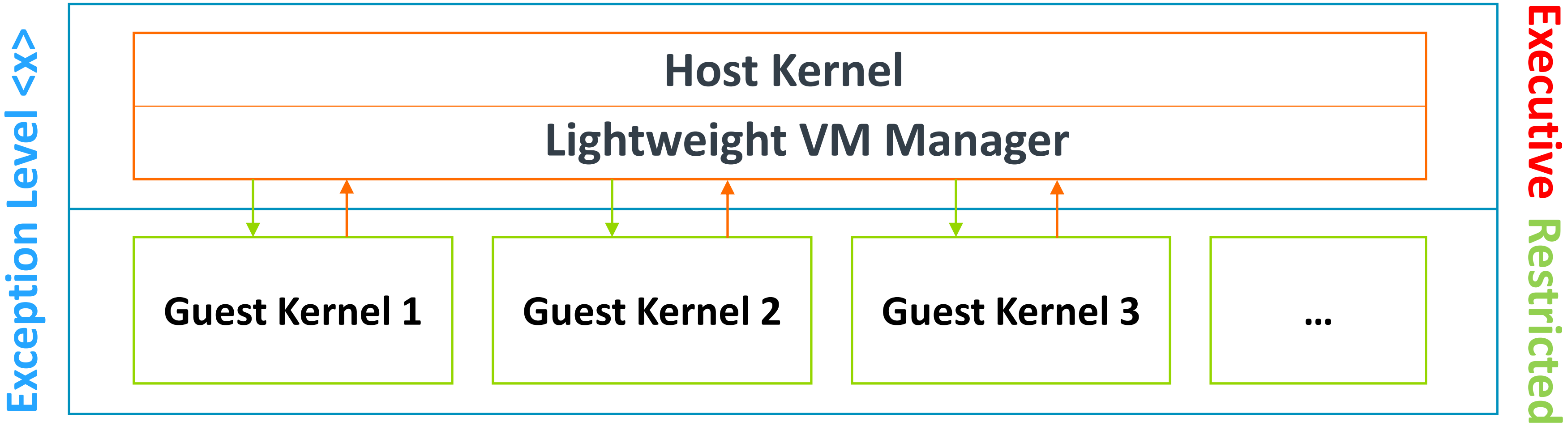
With the help of the community we aim to build a little prototype and measure performances, if they show a positive trend we will eventually look at porting a mainstream application to this model to demonstrate do-ability.

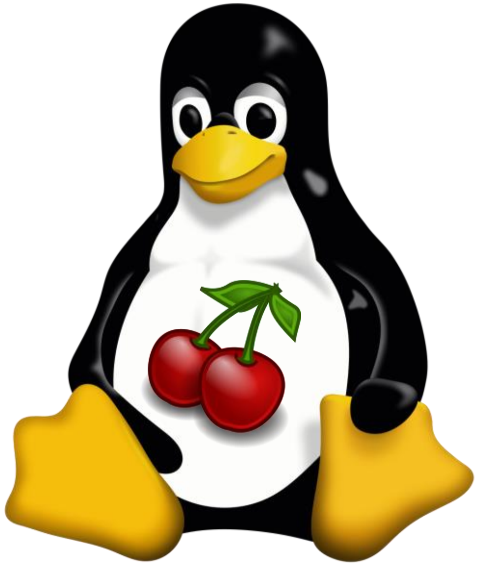
Status: We started the investigation in June 2023, it is the right time to join and help.



Morello Linux – Lightweight VMs (1/2)

HELP WANTED!!!





Note: ∃ x/x ∈ {1,2}

Morello Linux – Lightweight VMs (2/2)

HELP WANTED!!!

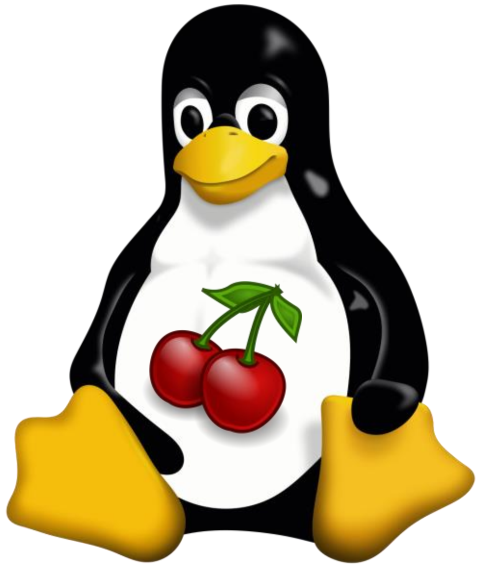
The idea is to leverage Executive and Restricted state to run a Host and a Guest kernel at the same Elx.

We know that this is not achievable with the current version of Morello architecture.

The purpose of the investigation is to find what is missing in the Morello Architecture to make this use-case possible.

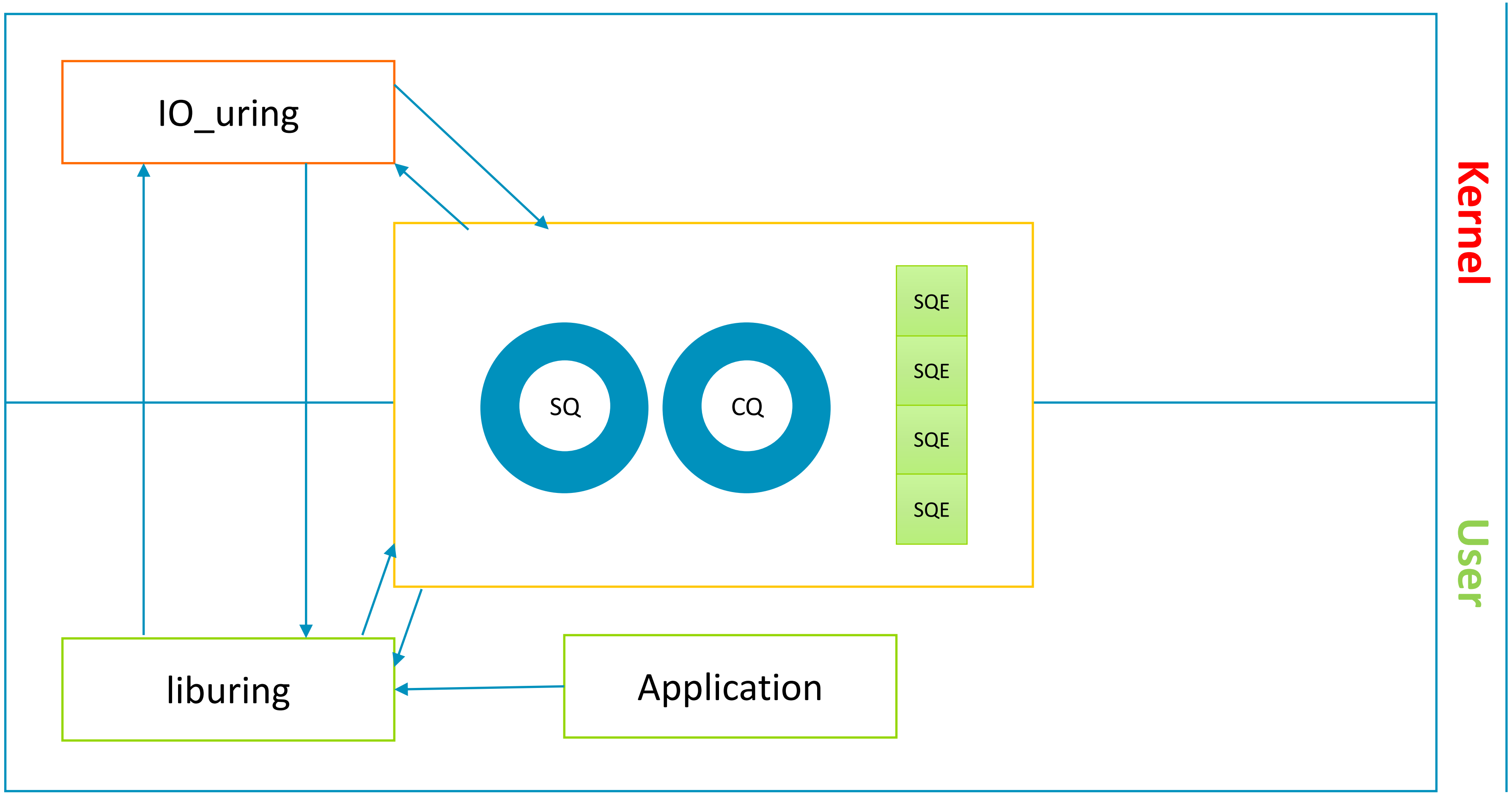
In doing so, with the help of the community, we aim to use pKVM as a reference and to try as a long term goal to emulate (where possible) the parts of the architecture that are missing.

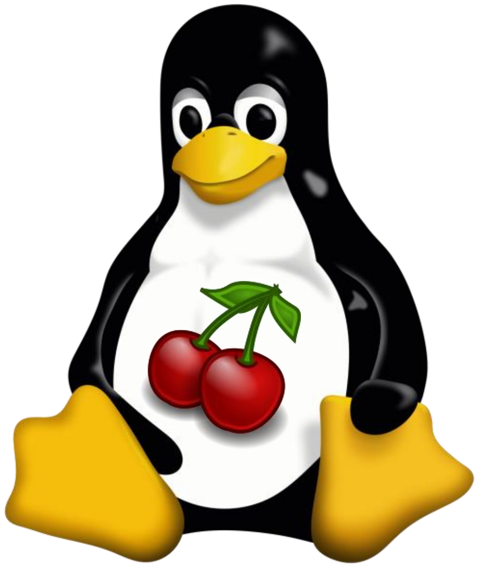
Status: We started the investigation at the end of June 2023. If you are interested in the topic, it is the right time to join the effort.



Morello Linux – IO_uring

HELP WANTED!!!





Morello Linux – March 2022: Discovered an 0-day to exploit, CVE-2021-41073

HELP WANTED!!!

CVE-2021-41073 Details [0].

[0] https://chompie.rip/Blog+Posts/Put+an+io_uring+on+it+-+Exploiting+the+Linux+Kernel

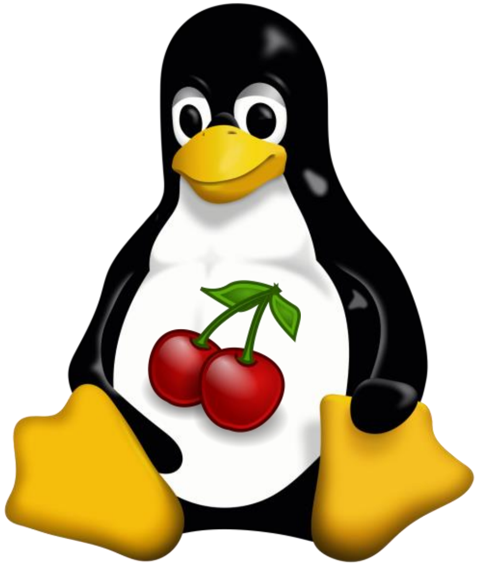
The exploit takes advantage of a use-after-free and uses it in a combination with an eBPF program attached to a socket to gain root privileges on the system.

This and other uncovered exploits made so that Google recently restricted the use of IO_uring across its Operating Systems:

"While io_uring brings performance benefits, and promptly reacts to security issues with comprehensive security fixes (like backporting the 5.15 version to the 5.10 stable tree), it is a fairly new part of the kernel. As such, io_uring continues to be actively developed, but it is still affected by severe vulnerabilities and also provides strong exploitation primitives. For these reasons, we currently consider it safe only for use by trusted components.“

Google's Blog Post: https://security.googleblog.com/2023/06/learnings-from-kctf-vrps-42-linux.html

Phoronix Report: https://www.phoronix.com/news/Google-Restricting-IO_uring



Morello Linux – Our Approach

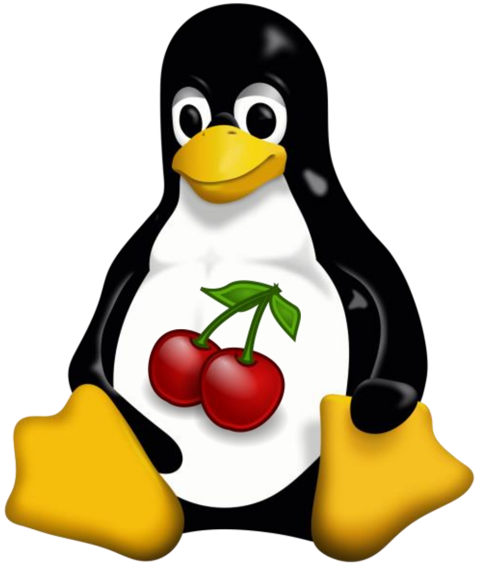
HELP WANTED!!!

Our approach: With the help of the community we aim to leverage Morello capabilities to provide security guarantees for IO_uring.

Our goal is to prove that it is possible to take advantage of the performance benefits provided by asynchronous system calls in a safe way.

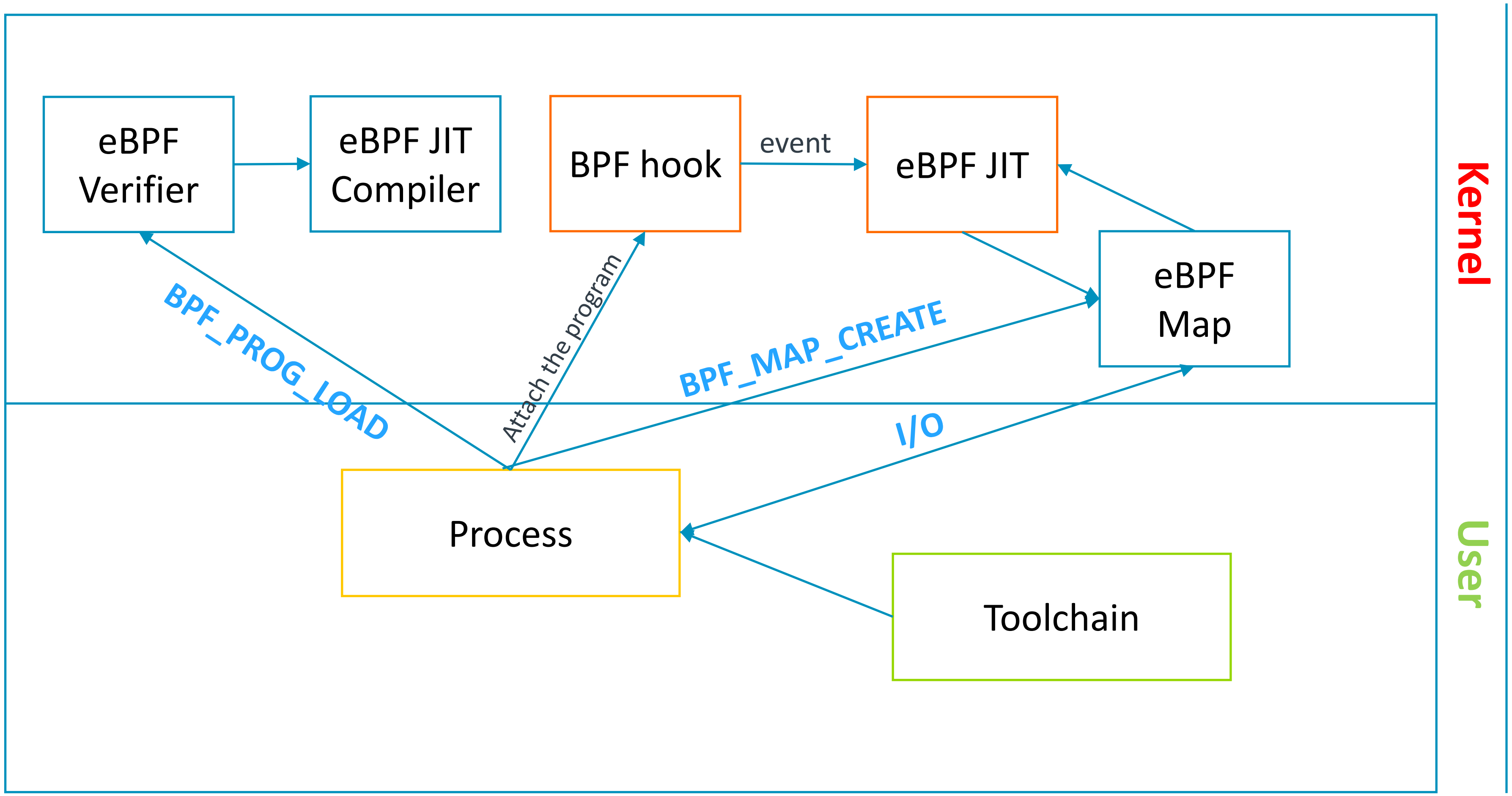
Status: We started the investigation more then 6 months ago and the initial bits are currently merged in our Morello Kernel.

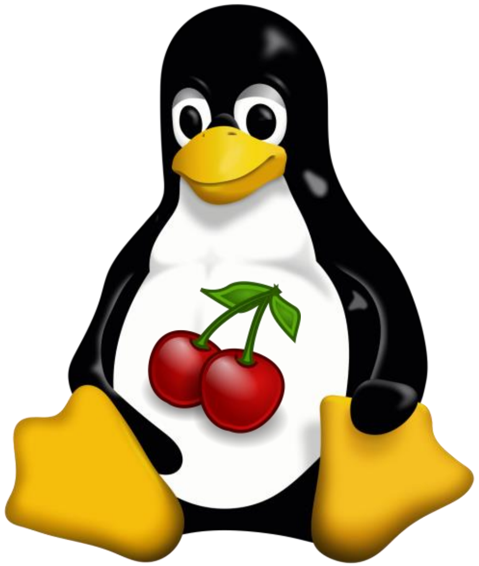
Note: A full solution to the problem highlighted by Google requires a Pure Capability Kernel and means to detect temporal safety programmatical errors to systematically detect the vulnerability, which is a perfect opportunity to join the effort and contribute to the development.



Morello Linux – eBPF

HELP WANTED!!!





Morello Linux – July 2021: Discovered a vulnerability, CVE-2021-3490

HELP WANTED!!!

CVE-2021-3490 Details [0].

[0] https://chompie.rip/Blog+Posts/Kernel+Pwning+with+eBPF+-+a+Love+StoryThe

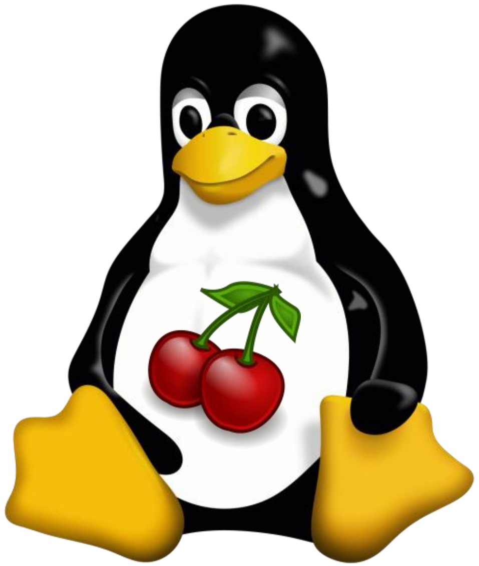
eBPF ALU32 bounds tracking for bitwise ops (AND, OR and XOR) in the Linux kernel did not properly update 32-bit bounds, which could be turned into out of bounds reads and writes in the Linux kernel and therefore, arbitrary code execution.

Our approach: With the help of the community we aim to harden the eBPF interface first and then focus on the JIT engine.

This is a way to improve performances through security, because would allow to use eBPF safely [1].

[1] https://security.googleblog.com/2023/05/introducing-new-way-to-buzz-for-ebpf.html

Status: Initial patches for the interface support are out on the list and we are in the process of shifting focus towards the eBPF JIT engine. If you are interested in eBPF this is the opportunity to help to enance security in this area.



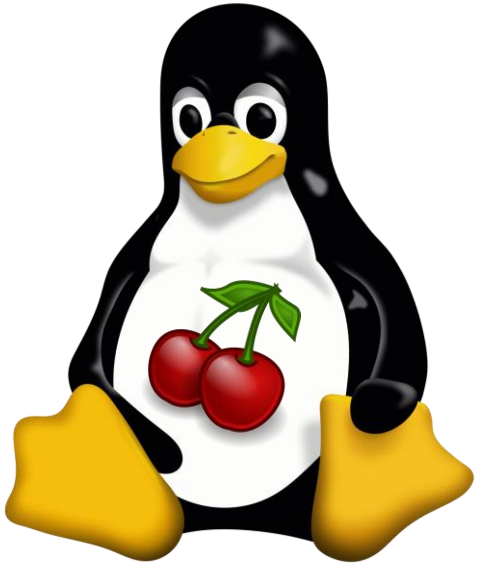
Morello Linux

Build a Morello Community

Morello Linux – Build a Morello Community

- **Goals:**
  - Define a lightweight process.
  - Simplify the submission of new features for the kernel.
  - Provide a common place to access the information.
  - Make everyone feel part of the same Morello community.

- **Main Focus:**
  - Mailing List
  - Common Task Tracker
  - Public CI



Morello Linux – Mailing List

- A public **Mailing List**: linux-morello@op-lists.linaro.org
- The mailing list is considered as the center of the community life.
- The mailing list purpose is to discuss:
  - New patches submitted for review.
  - New ideas for enhancing the Morello Linux Kernel.
  - Bugs and feature requests.
- All the activities of the morello kernel team are fulfilled through the public mailing list.
- All the contributions are expected to come through the public mailing list.

Morello Linux – Task Tracker

- A public **Task Tracker**:
  - https://git.morello-project.org/groups/morello/kernel/-/epics
- The purpose of the Tracker is to:
  - Create new tasks that impact the development of the kernel.
  - Give visibility to the tasks to the members of the Morello Linux Kernel community.
  - Define and identify tasks that are assigned to the members of the Morello Linux Kernel community.
  - Give the possibility to any member of the community to identify the status and a point of contact (PoC) for any given task.

Morello Linux – Public CI

- A public **Continuous Integration** (CI):
  - https://git.morello-project.org/morello/kernel/linux/-/pipelines
- The purpose of the CI is to:
  - Verify the status of one or more patches before merging them.
  - Build confidence in the delivered code base.
  - Decouple the consumption of the Linux kernel from its development.
- Test suites considered currently for inclusion in the CI for the Morello Linux Kernel are:
  - Kselftest
  - LTP (Linux Test Platform)
  - Musl tests.
- The plan is to extend the CI in future as part of the Morello Linux Kernel Development.

Morello Linux – Other Mailing Lists

- Morello-flavored Linux Test Project (LTP) discussions: linux-morello-ltp@op-lists.linaro.org
- Morello Linux Kernel CI Reports: linux-morello-ci@op-lists.linaro.org
- Morello Linux Distros Discussions: linux-morello-distros@op-lists.linaro.org

- **Note:** In future we would like to extend this model and what we learned to other components of the Morello Project.

Morello Linux – Thank you!

✉ vincenzo.frascino@arm.com

https://twitter.com/fvincenzo

Thank You

Danke

Merci

谢谢

ありがとう

Gracias

Kiitos

감사합니다

धन्यवाद

شكرًا

ধন্যবাদ

תודה