TinyContainer

Container Runtime Middleware Enabling Multi-tenant Microcontrollers with Built-in Security

WiSec 2026

9th ACM Conference on Security and Privacy in Wireless and Mobile Networks

June 30-July 03, 2026

By Bastien Buil (Orange Research & Cnam), Chrystel Gaber (Orange Research), Samuel Legouix (Orange Research), Emmanuel Baccelli (Inria) and Samia Bouzefrane (Cnam)

https://github.com/Orange-OpenSource/RIOT/tree/tinycontainer/examples/advanced/tinycontainer

Manage software containers on microcontrollers: Why? And how?

How do we…

Personalize logic for individualized customer experience?
Have third-party code for customizing user experience?
With continuous deployment between cycles of new hardware rollouts?
Without maintaining several firmwares in parallel?

➞ Need containers on microcontrollers!

But there are caveats…

What if a malicious container collects data without the user knowing?
What if a malicious container changes a sensor configuration thereby poisoning data collecting by another container?

➞ Need to check legitimacy of resources usage!

Even more caveats on microcontroller…

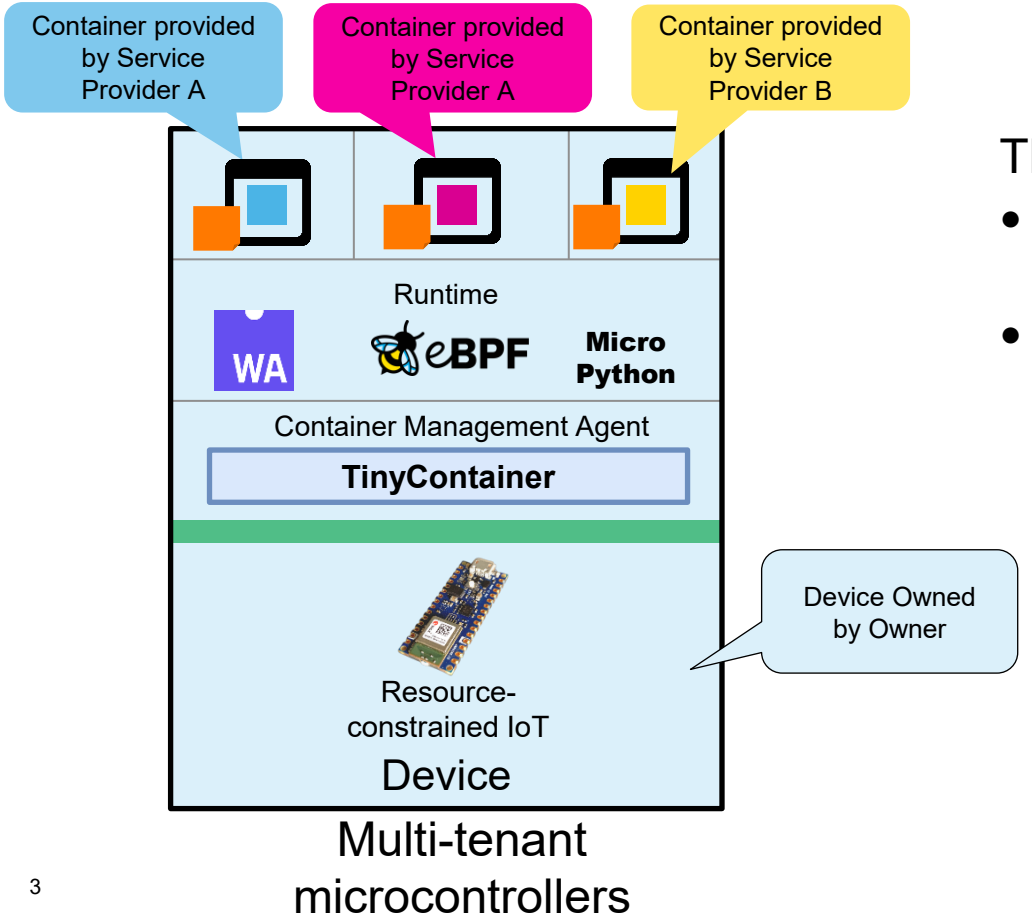
How can container-like system run on microcontrollers?

How to perform access control efficiently?

➞ Need built-in optimized mechanisms!

Microcontrollers
Between 256 kB & 512 MB Flash
Between 64 kB & 256 MB RAM

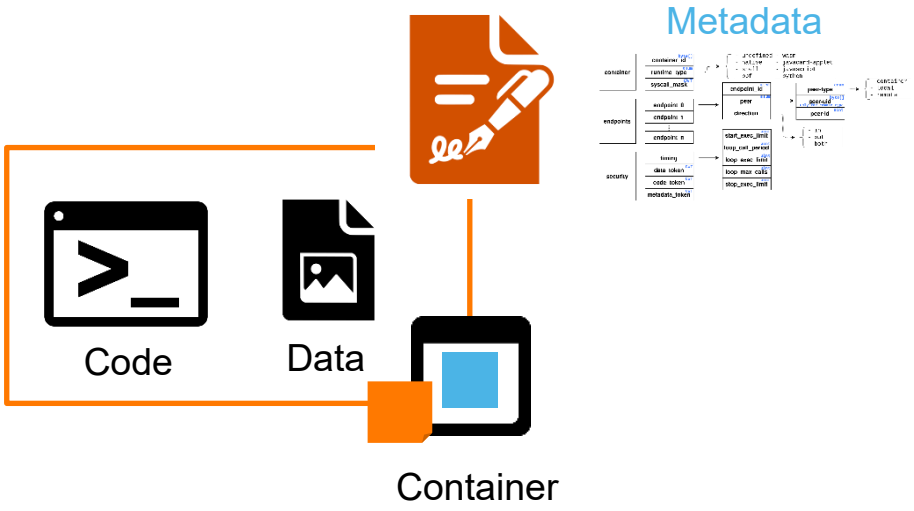
The target: secure multi-tenant microcontrollers



The device can be:

- multiple isolated containers, 1 stakeholder
- **multiple containers, multiple stakeholders**

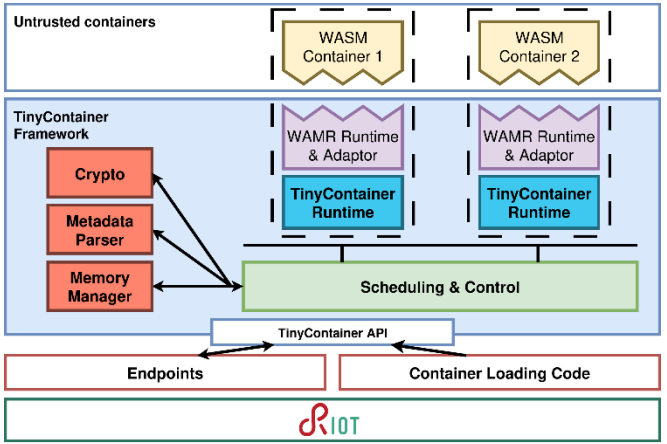
What is a container ?



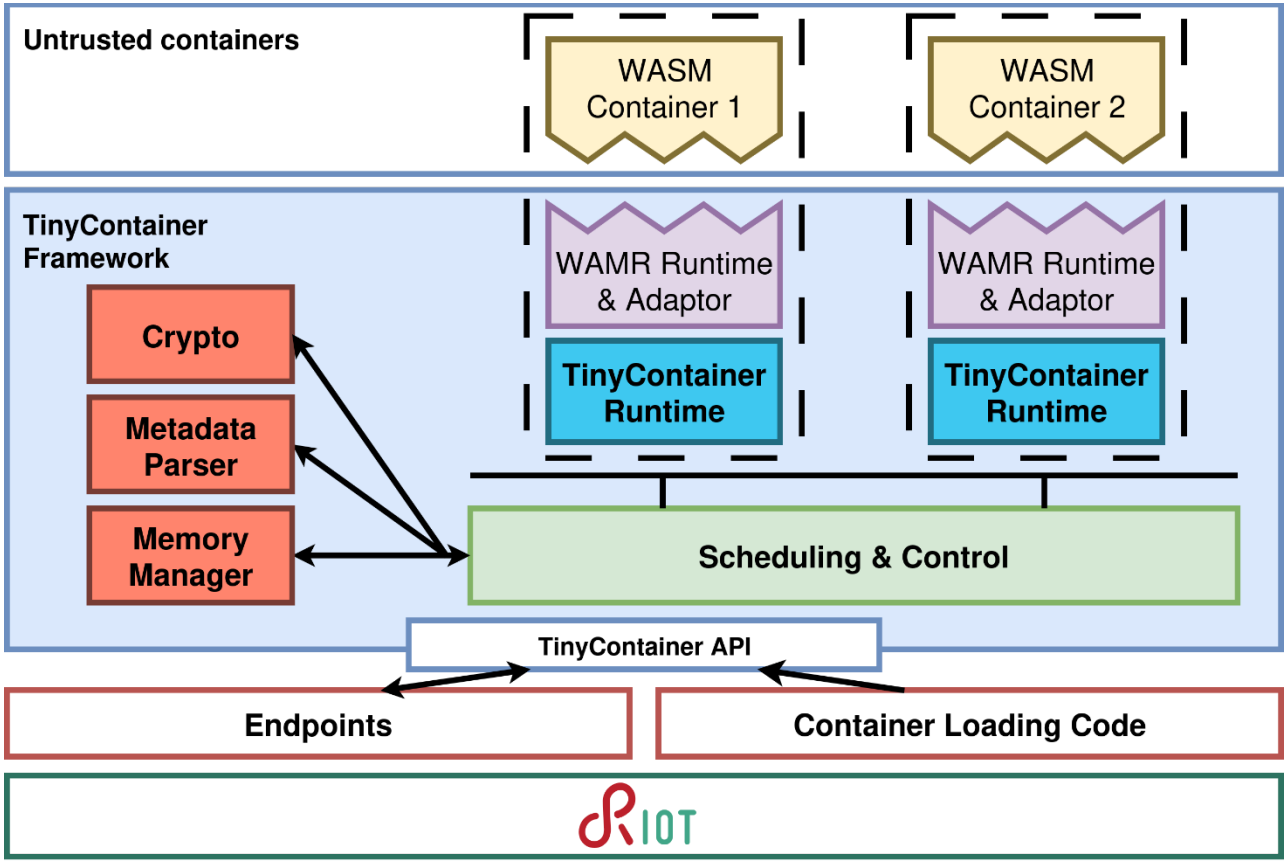
TinyContainer: Access Control & Scheduling for Virtual Machines Sharing Resources on Microcontrollers

Proposition: TinyContainer, a lightweight software container management framework designed for multi-tenant microcontrollers.

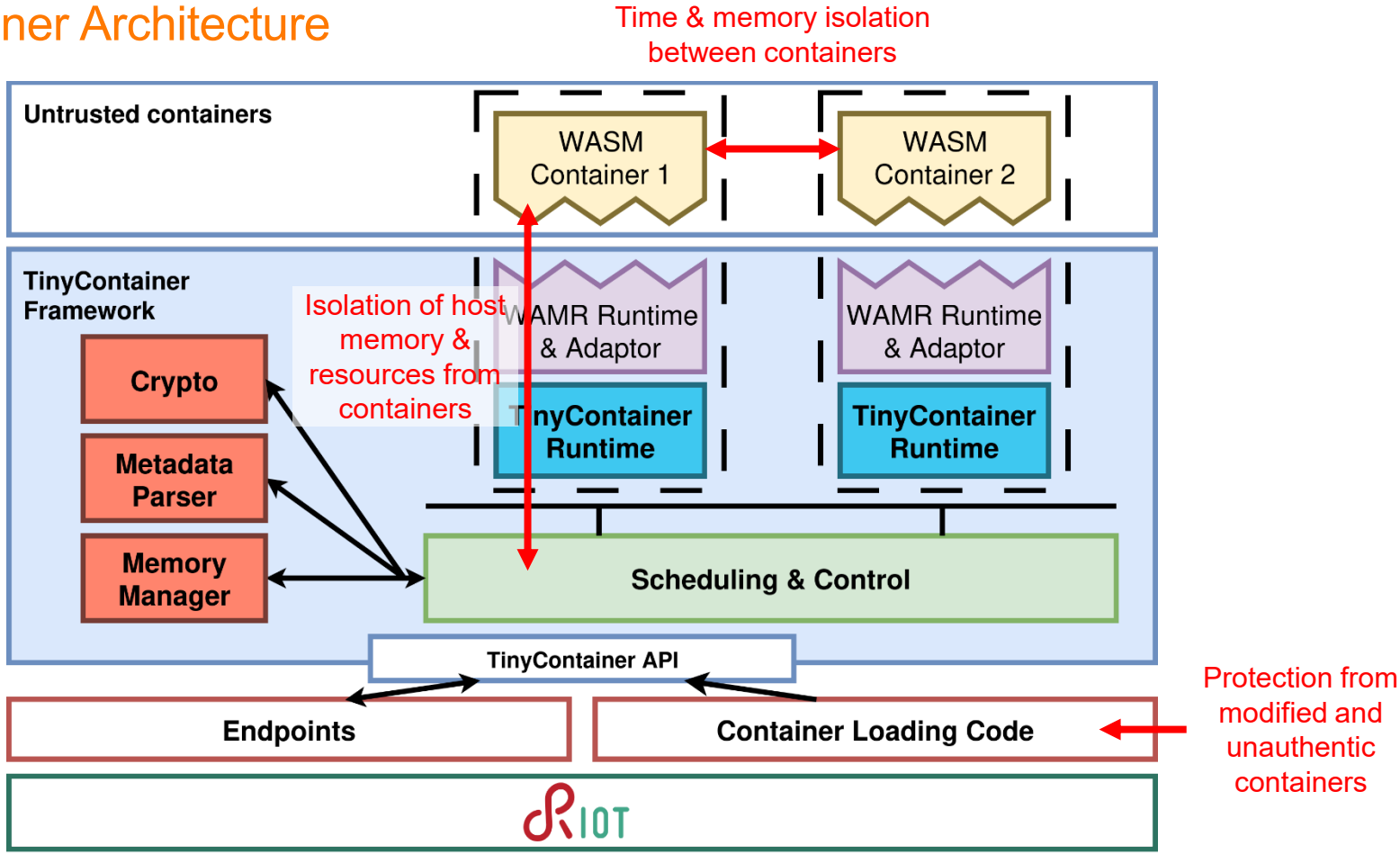
1. per-container configurable scheduling with watchdog system,
2. fine-grained access control to host resources with an endpoint system,
3. metadata with built-in security for easy remote configuration and deployment
4. compatible with multiple runtimes via a runtime abstraction layer, demonstrated with CS4WAMR



TinyContainer Architecture

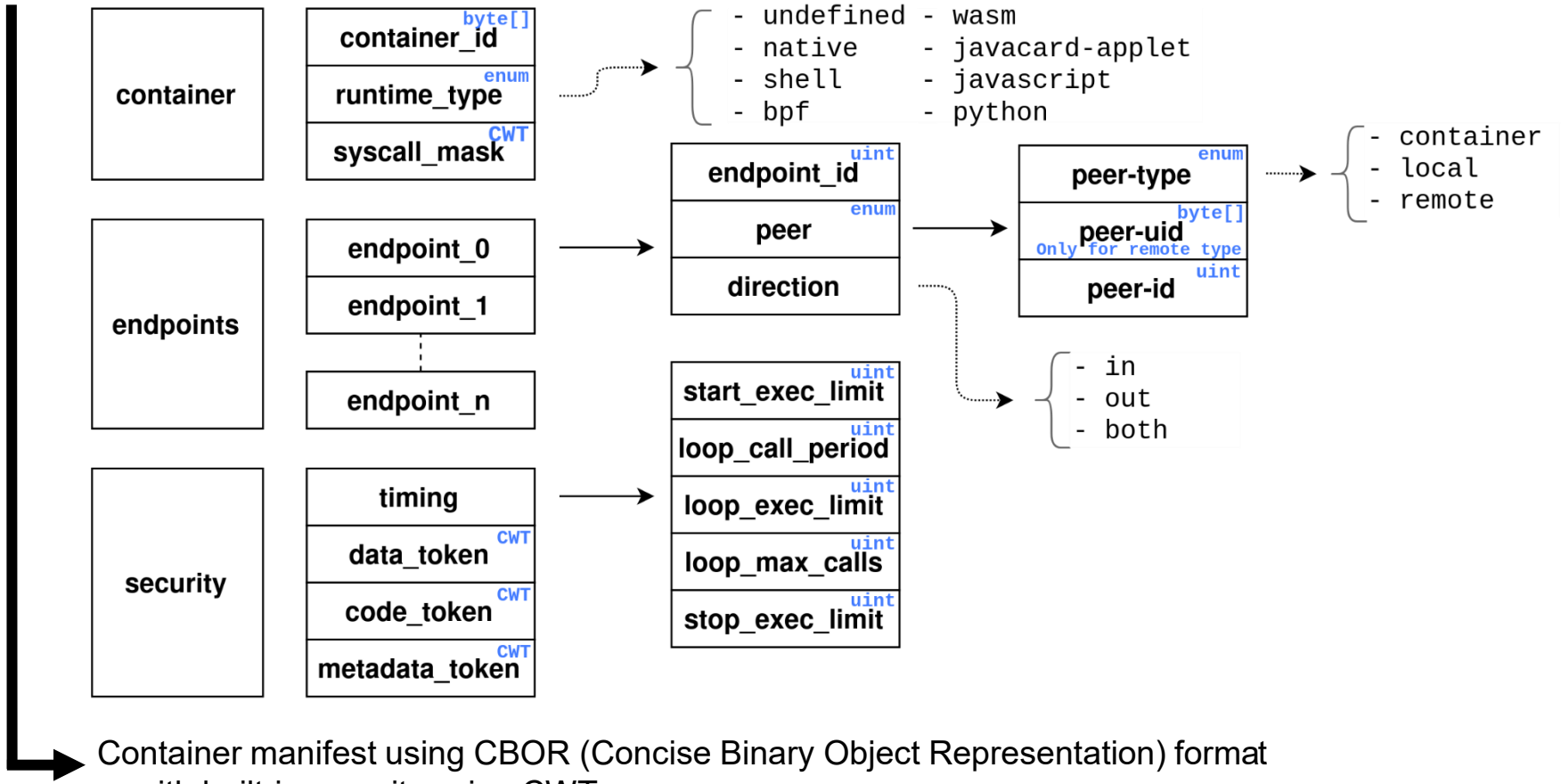


TinyContainer Architecture



Internal working of TinyContainer

TinyContainer metadata



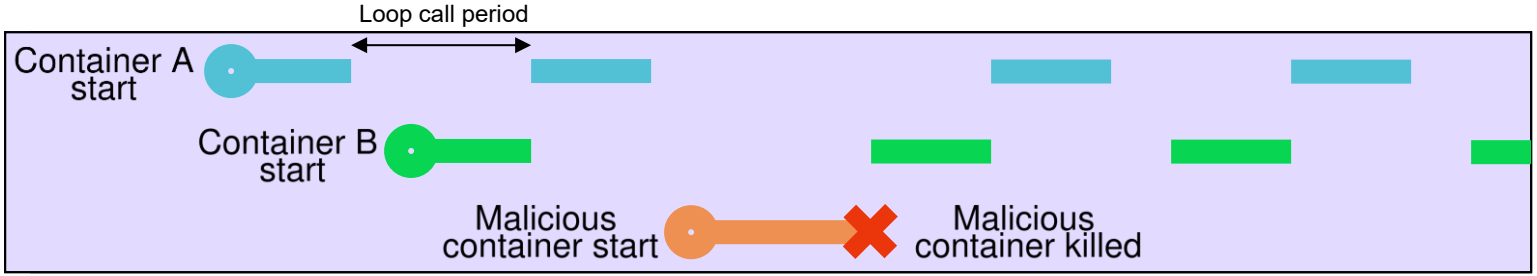
Container manifest using CBOR (Concise Binary Object Representation) format with built-in security using CWT

TinyContainer Scheduling system

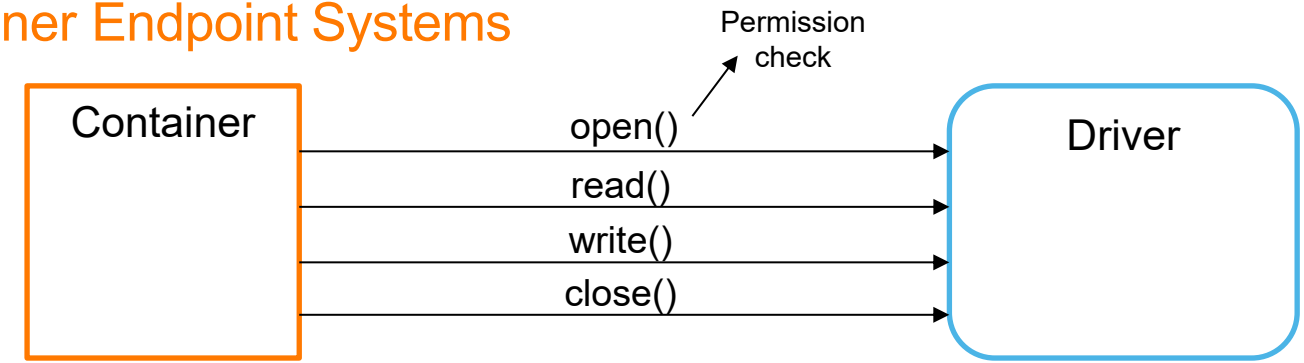
- Each container has three entry points: *on_start()*, *on_loop()*, and *on_stop().*

- The system enforces customizable per-container:
  - time and call limits to prevent resource monopolization,
  - call interval to on_loop function to wake up of containers.

- If time limit is exceeded, a watchdog intervenes, ensuring fair resource sharing and system responsiveness.



TinyContainer Endpoint Systems



- Host can implement endpoints by implementing a driver containing these functions
- Similar to Unix input-output system calls with:
  - open() function
  - read() function
  - write() function
  - close() function
- Allow fallback mechanism if the container does not have permission
- Permissions are loaded when parsing the metadata manifest file and control access to endpoints

Can we run TinyML using TinyContainer?

TinyMLaaS endpoint

Issue:

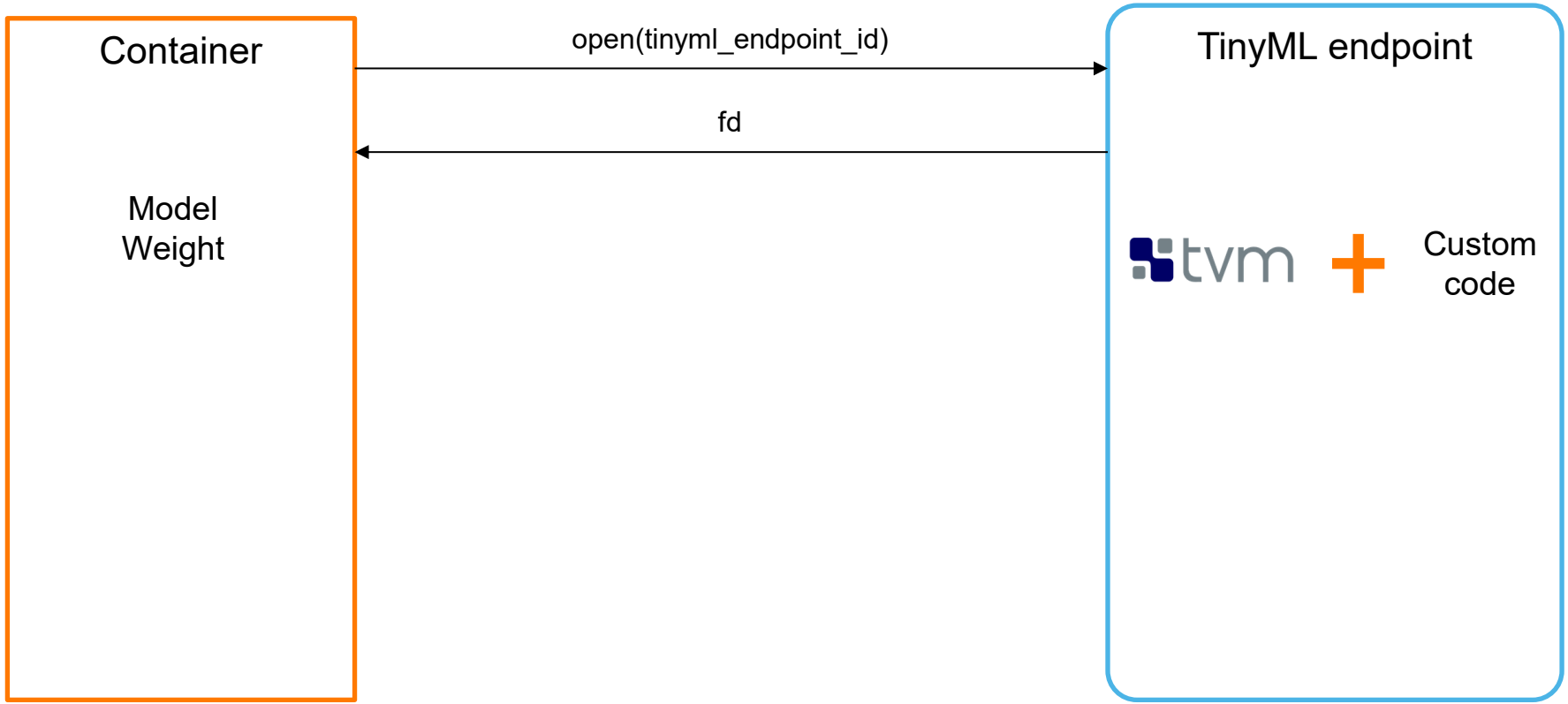
- TinyML, like any code, can contain malware
- Running TinyML in containers without Ahead-of-Time compilation is not feasible, due to massive performance drop
- Alternative like WASI-NN, requires using libraries like LiteRT which uses a lot of memory, thus not adapted to most constrained devices

Proposition: Can the host optimize one type of model and allow container to run their model through TinyContainer endpoint

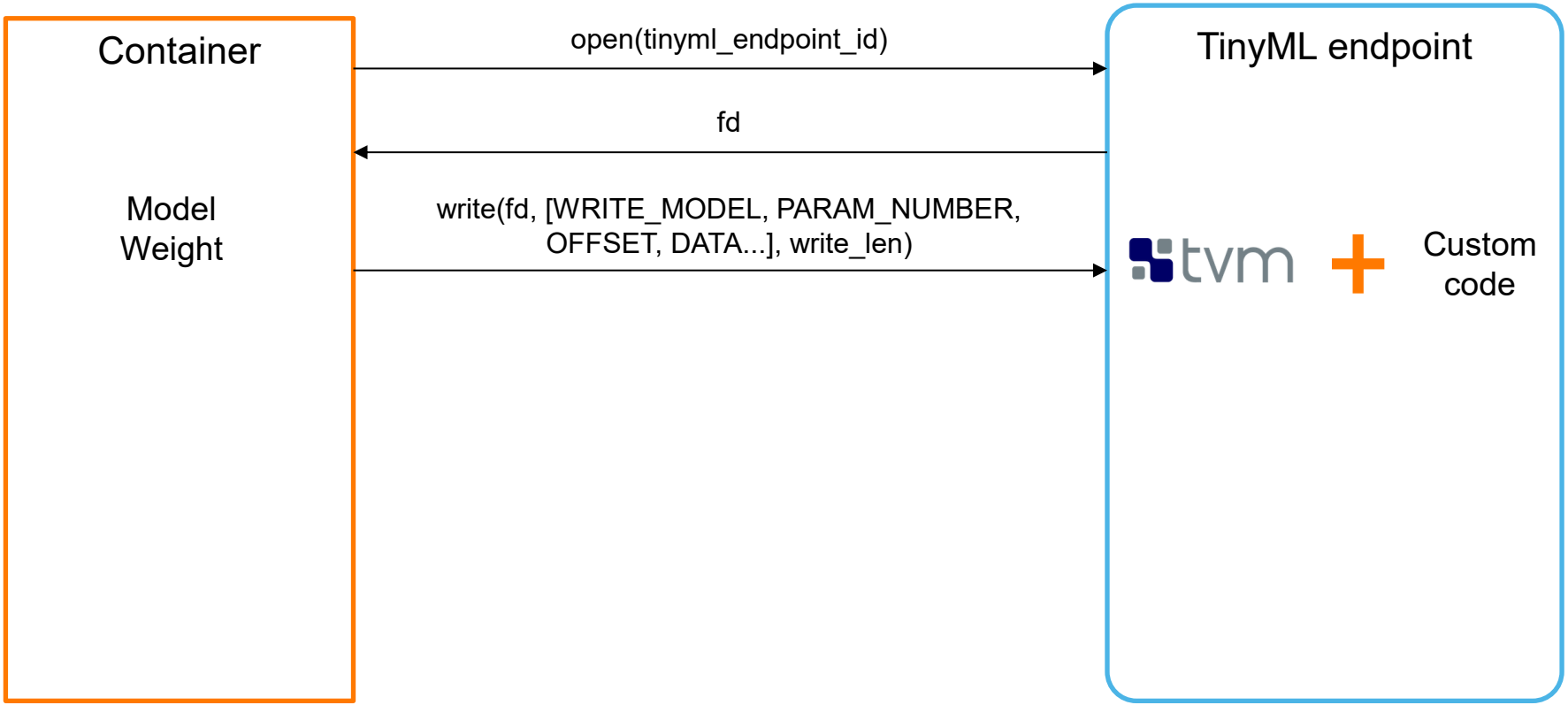
1) Host announces a model structure
   A. Provide a base model to fine-tune
   B. Give the structure (e.g. multi-layer perceptron): require training from scratch a model, optionnaly with Knowledge distillation

2) Provider deploys a container with model weights and logic to use the model endpoints

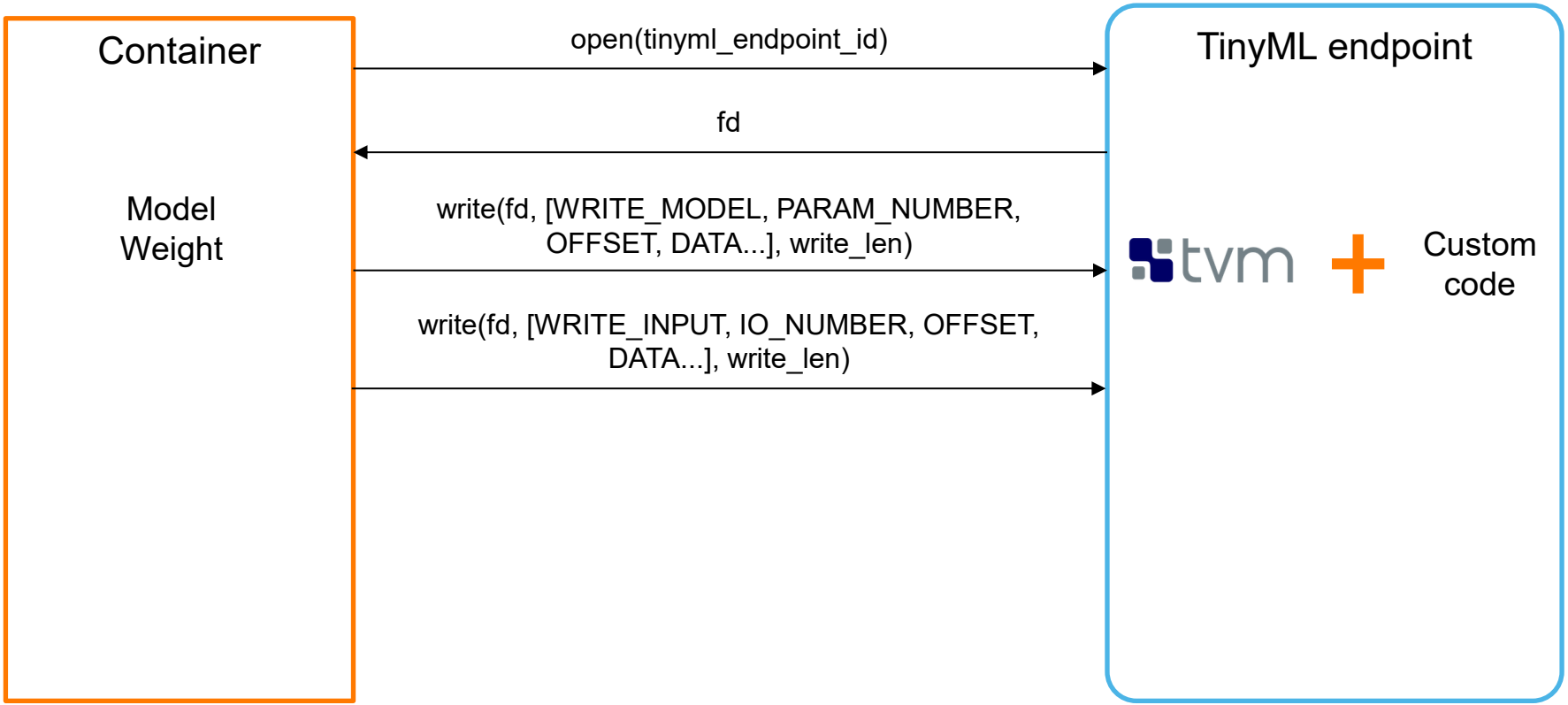
TinyMLaaS endpoint



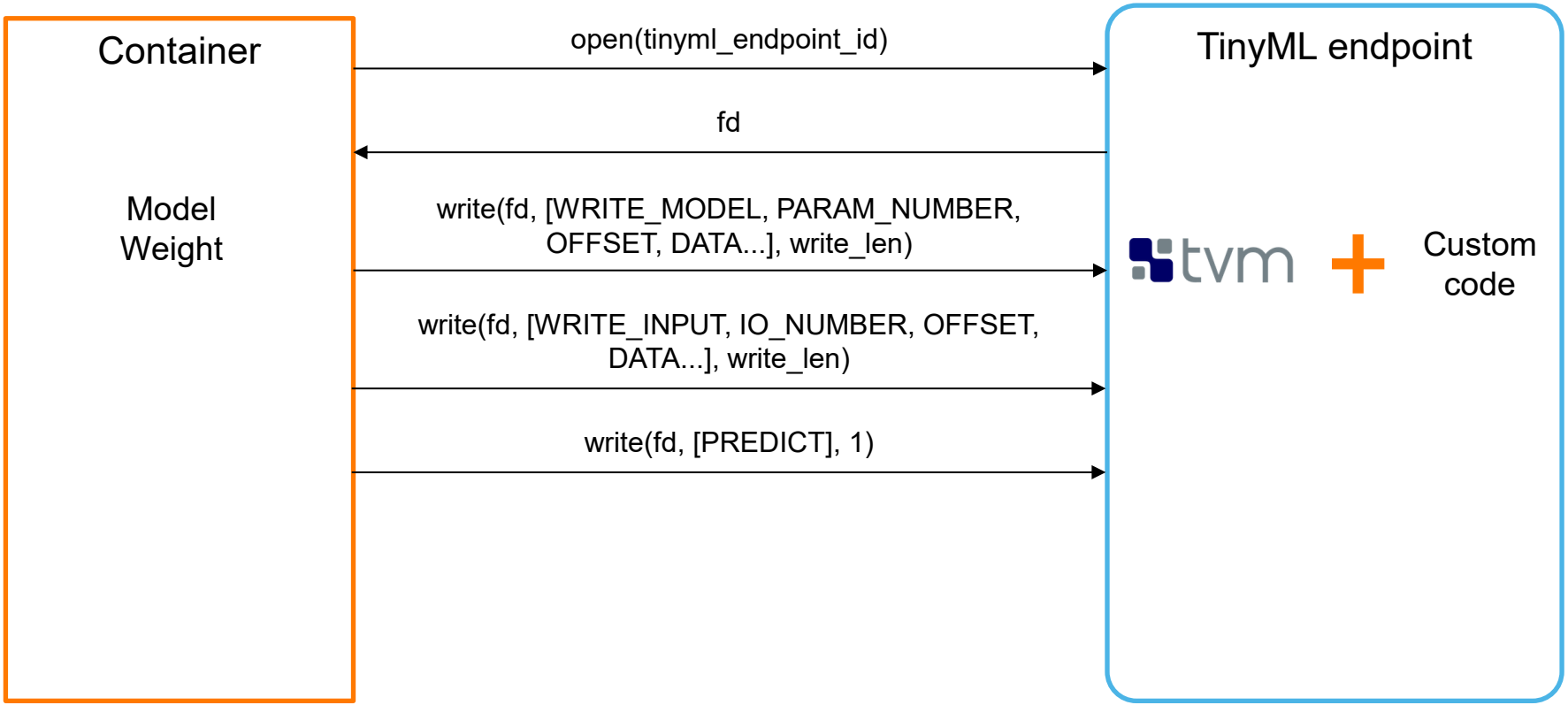
TinyMLaaS endpoint



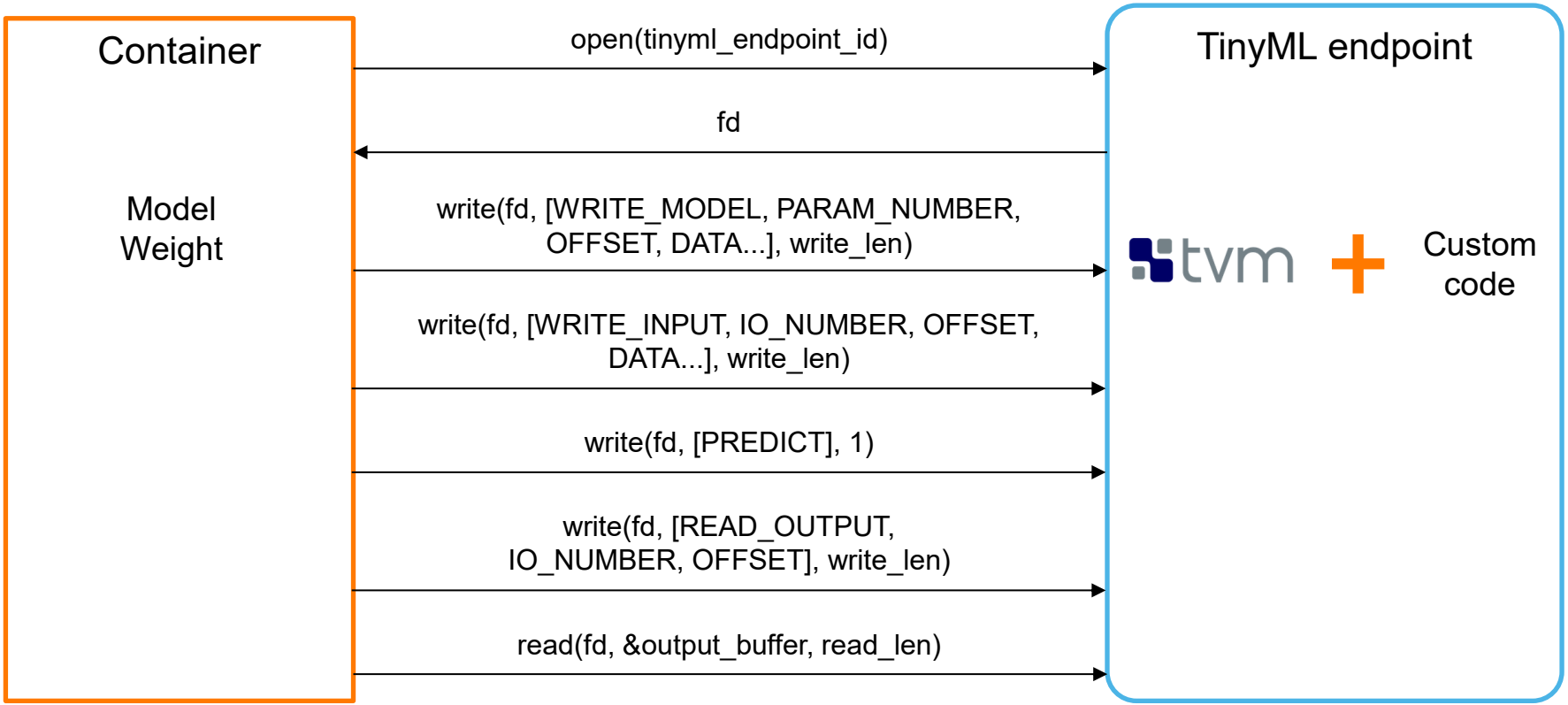
TinyMLaaS endpoint



TinyMLaaS endpoint



TinyMLaaS endpoint



Benchmarks

Overhead for TinyML computation

	Inference time (in ms) with DS-CNN Small model [29]	
	Arduino Nano 33 BLE	nRF9160-DK
In-container WAMR AoT	442	497
In-container WAMR Interpreter	48 324	48 130
TinyContainer TinyML native endpoint	512	507

The inference time is the measured time of the execution of the container function running inference or the container function calling inference endpoint for TinyContainer. It includes all the inference time and the communication cost, and the Wasm container function starting cost. For TinyContainer part, it does not include the loading of the model from the container to the driver.

Compute overhead

Two majors overheads:

- Loading container metadata and code
  - ➢ Very costly on board without CryptoCell
- Accessing endpoints

Performances of running a polling container interacting with one endpoint

Runtimes	Execution time (in ms)	
	Arduino Nano 33 BLE	nRF9160-DK
WAMR Interpreter		
Load container	1.61	1.67
Container execution	0.91	0.98
CS4WAMR		
Load container	2.21	2.16
Container execution	0.97	0.99
TinyContainer with CS4WAMR (without cryptography)		
Load container metadata	19.50	19.33
Load container	3.54	6.04
Container execution	63.23	60.10
TinyContainer with CS4WAMR		
Load container metadata	91.15	8 810.16
Load container	3.61	6.05
Container execution	74.0	63.54

Other limitations of current implementation

Currently missing features:

- Inter-container communication
- More wake-up mechanisms
- Multiple runtimes at the time

Conclusion

Conclusion

Tinycontainer is a middleware for runtimes on microcontrollers:

- which enables dynamical management of containers
- while providing built-in:
  - Permission management per container,
  - Dynamically loaded metadata,
  - Configurable execution scheduling per container.

Thank you

WiSec 2026

9th ACM Conference on Security and Privacy in Wireless and Mobile Networks

June 30-July 03, 2026

By Bastien Buil (Orange Research & Cnam), Chrystel Gaber (Orange Research), Samuel Legouix (Orange Research), Emmanuel Baccelli (Inria) and Samia Bouzefrane (Cnam)

TinyContainer

https://github.com/Orange-OpenSource/RIOT/tree/tinycontainer/examples/advanced/tinycontainer