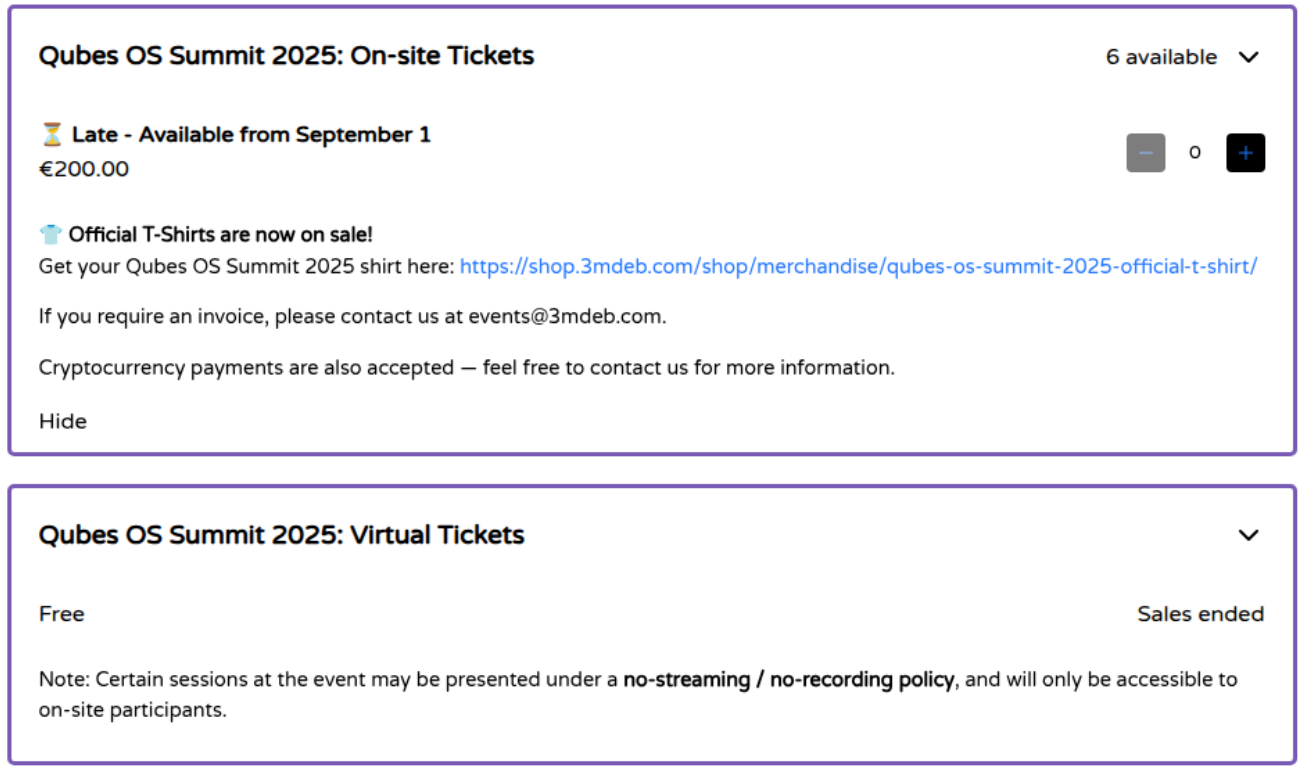
👋 Dasharo User Group #11 🎉 and Developers vPub 0x10 🍻

👋 DUG#11 Closing Remarks 👋



https://events.dasharo.com/event/2/qubes-os-summit-2025

https://hardwear.io/netherlands-2025/training/mastering-uefi-secure-boot-and-intel-root-of-trust-technologies.php

EDK II on ARM: Booting EDK II on Odroid M2

2025-09-18 18:00–18:20 20:00-20:20 (Europe/Warsaw), Dasharo Developers vPub

This is a talk about the challenges of booting EDK II on an ARM platform and the overall state of EDK II on ARM. I will be describing the process of porting a new platform using pre-existing SoC support code and give a short demo of an early prototype of EDK II based firmware for the Odroid M2.

Michał Kopeć

Michał has been at 3mdeb for 4 years and has worked on x86 boot firmware for laptops, desktops and network appliances, as well as developed Linux kernel drivers and firmware security features. He has an interest in all things open source and enjoys hacking and/or bricking hardware in his free time.

BSD MAC LLM UI

2025-09-18 19:00–19:20 21:00-21:20 (Europe/Warsaw), Dasharo Developers vPub

This talk introduces BSD3 MAC LLM UI, a tiny, auditable LLM chat interface built for teams that value isolation, predictability, and a minimal attack surface. Written in C with suckless coding standards and released under the BSD 3-Clause license, the project provides a no-JavaScript HTML/CSS web UI and an optional GTK/Qt local GUI. On the backend it can route prompts to an OpenAI-compatible API or run fully offline via TensorRT-LLM (v0.21.0) C++ bindings—while remaining safe to deploy in MAC and security by compartmentalization environments (OpenBSD, Linux OS; OpenXT/Qubes Hypervisor).

We'll cover the core design: a small HTTP/1.1 server, stateless form posts (no DB), strict caps/timeouts, and hardening with pledge(2) on OpenBSD and seccomp on Linux. We'll show deployment patterns for localhost-only, WireGuard segments, and Tor hidden services (works in Tor Browser with JS disabled). For developers, we'll walk through the single-binary build with Makefile knobs (HAVE_TRTLLM, WITH_GTK/Qt, TLS_BACKEND), compile-time configuration via config.h, and how to switch between networked and no-network modes (including future design for Qrexec/HMX to an inference VM). Attendees will leave with a practical template for building security-first, low-overhead LLM front ends that fit air-gapped, offline, or highly regulated stacks—without dragging in a mountain of dependencies.

Arthur Rasmusson

Arthur Rasmusson is a kernel programmer, virtualization, and machine learning engineer specializing in GPU drivers, and AI inference. Arthur Rasmusson's experience spans open and close source software development where he has held the roles of Founding Engineer, Principal AI Engineer, and Machine Learning Engineer. As Founding Engineer of Arc Compute, he authored open-source documentation, and GPU virtualization software for the Open-IOV project. At Cohere, his work on Just-In-Time-Inference and Just-In-Time-Training reduced auto-scaling latencies by orders of magnitude by introducing the use of GPUDirect Storage API for real-time scaling of hardware allocations in hybrid training and inference environments to meet customer demand and maximize utilization. Prior to joining Weka he introduced Paged Attention over RDMA (PAoR) to the open source AI community used distributed filesystems in GPU clusters to improve LLM performance by saving wasted computational resources spent on redundant cache data generation in open-source inference servers delivering orders-of-magnitude performance gains at scale. At Weka, as Principal AI Engineer, he authored the open-source software behind AI "token warehouses" with upstream source code contribution to the open-source TensorRT-LLM project adding the "KV Cache GPUDirect Storage" feature, and implementing Python native support for GPUDirect Storage APIs used in LMCache for the vLLM ecosystem. Arthur Rasmusson's work bridges low-level systems concepts with high-performance data paths for AI.

The adventure of porting a retail AMD server board to coreboot

2025-09-18 20:00–20:20 22:00-22:20 (Europe/Warsaw), Dasharo Developers vPub

The talk will describe the first-hand experience of porting an AMD server board to coreboot and enabling a server processor family support in the project as well. Technical challenges and AMD OpenSIL integration will be discussed. A short demo of booting Gigabyte MZ33-AR1 to UEFI payload setup will be shown.

Michał Żygowski

Michał Żygowski is a versatile engineer with a strong focus on system firmware. Works as a firmware engineer at 3mdeb . Active contributor of coreboot and other open-source projects. Core coreboot developer, maintainer of Braswell SoC, PC Engines, Protectli and MSI platforms. Loves travelling and attending conferences, which actively speaks on. Mainly interested in the firmware, security and advanced hardware features.

TEST + CHOOSE + KNOW

YOUR FIRMWARE

DASHARO

- **Test**: Dive into our latest releases, test them out, and share your feedback. Your experiences refine our roadmap. 🧪
- **Choose**: Choose to be an active part of our community. Your engagement shapes Dasharo's evolution. 🛠️
- **Know**: Stay informed and share your knowledge. Together, we deepen our understanding and create a robust firmware solution. 🎓

Your Actions Matter 🌟 Thank You 🙏

- **Spread the Word**: Help more people discover, test, and choose Dasharo. 🗣️
- **Contribute**: Your code, documentation, and ideas are the building blocks of Dasharo's growth. 👩‍💻👨‍💻
- **Engage**: Join our discussions, forums, and DUG meetings. Every interaction enriches our community. 🤝

Let’s Switch to vPub