

Porting Android Automotive on Xen

Leo Yan

Linaro Performance and Enablement

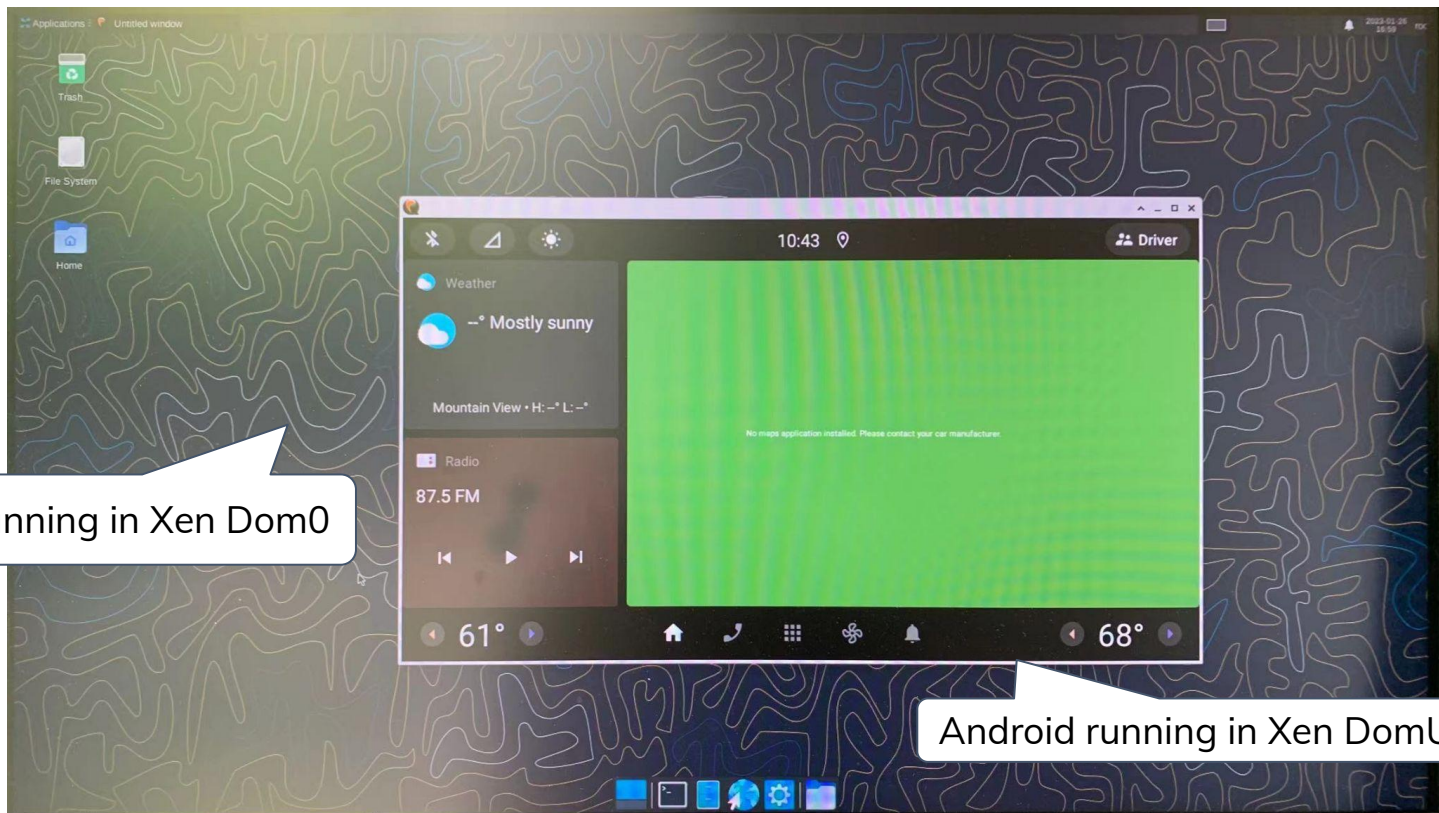


Introduction

- As a part of automotive project, the goal is to identify the gaps associated with running Android Automotive OS in a virtual machine using the Xen hypervisor.
- The objective was to leverage the Android Cuttlefish VM as a Xen hypervisor guest with using virtio based virtual device emulations. That involved:
 - Host is Yocto based distro: [TRS](#) or SoC vendor's distro (includes Xen, Linux kernel, and QEMU, etc)
 - Guest is Android Open Source Project (AOSP), built on master branch with aosp_cf_arm64_auto-userdebug* target
 - The proof of concept is in part based on work done by EPAM
- Tested on 3 hardware platforms:
 - [ADLink AVA Platform](#) (Ampere Altra SoC, 32 Arm Neoverse N1 CPUs)
 - [Telechips Dolphin5 automotive platform](#) (Arm Cortex-A76 x 4 + Cortex A55 x 2)
 - [Rockpi 4b](#) (Arm Cortex A72 x 2 + Cortex A55 x 4) (Jun Nie)

* It was renamed as aosp_cf_arm64_auto-trunk_staging-userdebug in Oct, 2023

Screenshot :-)



Linux running in Xen Dom0

Android running in Xen DomU

Current status

- Support Xen virtual machine in Android U-boot [[patches](#) - merged]
- Support Xen fundamentals and frontend drivers in Android common kernel [[patch](#) - no plan for upstreaming]
- Support Xen virtual machine in Android Cuttlefish device folder (\$android/device/google/cuttlefish)
Set display finder mode as “drm” [[patch](#) - merged]
Support booting on Xen virtual block device [[patch](#)]
Fix console hvic0 regression [[patch](#)]
- Maintain [Xen](#) and [QEMU](#) repositories for support virtio devices on Xen / Arm64
- Known issue: Cuttlefish’s secure_env is absent on Xen, file system encryption, oemlock and other security services have been disabled.
- [Demo](#): Booting Android automotive OS on Xen virtual machine

Next steps

- Upstream Android Cuttlefish patches for better support Xen virtual machine.
- Support virtual devices with vhost based backends
 - Develop vhost user based backends in a hypervisor-agnostic way: [vhost-device](#)
RNG, I2C, GPIO, SCSI, sound (WIP), input (WIP), etc.
 - Vulkan support for virtio-gpu
- Any suggestions are welcome!

Thank you!