

Viability of NixOS on the PinePhone Pro, with GrapheneOS via Waydroid as an android compatibility layer for secure, open, and practical consumer use

Carolina Rivera

04/15/2024

Abstract

This report aims to analyze the process of installing NixOS (**NOS**) on the PinePhone Pro (Explorer edition) (**PPP**) with Waydroid as a compatibility layer to run android applications. Rather than using AOSP, I analyze the process of using a generic image of GrapheneOS (**GOS**) onto Waydroid for an experience granting additional security and privacy features, like the included hardened memory allocator. The purpose of this software stack is to provide end users with a transparent, secure, and private mobile operating system, with strong android app compatibility.

Introduction

The PPP is Pine64's flagship mobile smartphone, stated to be the fastest mainline Linux smartphone available on the market. It is the PinePhone's second major reiteration, mainly boasting hardware improvements. Despite being a refreshed device, there is a very long way to go before any mainline linux smartphone is ready for daily or consumer usage. Most notably, critical components of the device such as the bootloader, camera, and suspension range from being unimplemented, to being experimental with substantial bugs.

Nix is a purely functional package manager, where packages are treated like values would be in other purely functional programming languages. Each declared function does not result in side effects, allowing for Nix to have a substantial degree of reproducibility. Along with Nix there is NixOS, a GNU/Linux distribution built on top of the Nix package manager. Its biggest differentiation from Nix standalone is its ability to also build and manage configuration files through nix expressions, even down to kernel configuration.

Mobile NixOS (**MNO**) sits on top of NixOS, Nixpkgs, and Nix, with the ultimate goal of abstracting away the differences between mobile devices as is done on desktop computers. Basically, "NixOS, [but] on your phone".

GrapheneOS is a private and secure mobile operating system based on mainline AOSP, boasting substantial hardening enhancements to android such as their hardened malloc, proxies to insecure and privacy violating services like GNSS, and hardened libc.

Waydroid is a container based approach to booting a full android system on GNU/Linux through Linux namespaces, allowing the container and android to have direct access to needed hardware. This allows for substantial speed improvements without having to deal with virtualization.

This paper will also serve as a log of my exploits in trying to run said software stack on the PinePhone.

Steps in achieving the desired software stack

1. Gain an understanding of the PinePhone Pro, Waydroid, Nix, NixOS, GrapheneOS, and how they interact with each other.
2. Install Mobile NixOS onto the PinePhone
3. Setup a wayland compositor and sane configuration for development
4. Install Waydroid and run android through it's default configuration
5. Research and test the viability of running GrapheneOS for it's system image

Installation progress of Mobile NixOS

MNO on the PPP is only supported through the Tow-Boot bootloader installed onto the SPI, while the device comes with U-Boot. Tow-Boot is a user-friendly, and self proclaimed opinionated distribution of U-Boot. First I flashed Tow-Boot to a micro SD card to replace the installed U-Boot image on SPI. Once installed, I rebooted the device and noticed that there was no longer U-Boot's graphical interface, and that Sailfish OS was instead directly booted without further input, indicating that the installation has gone as expected.

Current issues

There have been significant issues with booting MNO on the PPP, and there are things that I can note to attempt diagnosing it. When booting MNO using a Hydra-compiled build made on April 15th, 2024, attempting to boot off the SD card would result in the LED indicator notifying a kernel panic. This was likely a result of having U-Boot rather than Tow-Boot, as the Mobile NixOS wiki page for the PPP states it can only boot with Tow-Boot installed onto SPI.

I tried again with Tow-Boot on the SPI, and an image built on April 19, 2024. Instead of a kernel panic, I managed to get significantly further into the boot, only for the indicator LED to flash white and yellow a few times until it gave up, then booting into the Sailfish OS installation. Having been able to successfully install Tow-Boot, this shows that the issue lies with the image used on MNO's PPP port.

Where now: Possible mitigations

1. Install the Nix package manager onto another distribution of GNU/Linux that's optimized for mobile devices. This would yield the benefits of the Nix package manager, while lacking the benefits NOS grants. We would also receive the benefits of basing our work on a stable, more mature project, allowing for a more seamless development environment for other goals, like porting GrapheneOS to Waydroid.
2. Debug and find a solution for the boot issue. This could eventually be sent upstream to be patched by whatever project is responsible for said issue. This may be the ideal solution, since MNO is a very new and slow moving project, also allowing the possibility to aid current maintainers.
 - a. This should eventually result in improved documentation
 - b. The issue may stem from the lead maintainer not being aware of new changes brought in by the latest PPP releases. I say so due to the PPP wiki page stating that there are firmware differences between certain revisions grouped by date of purchase.
 - i. The last commits to PPP's repository on MNO were from 7 months ago, despite the latest revision releasing 5 months ago in November 2023.