Thoughts on the "physically secure" ORWL computer

6-7 minutes

Several people, including some computer journalists, have asked me recently for an opinion on ORWL - “The First Open Source, Physically Secure Computer”. Below I provide a quick review of some of the features they boast about on their crowdfunding page (linked above and quoted below), then jump into more general conclusions and advice. They write:

> A battery-backed secure microcontroller (MAX32550 DeepCover Secure Cortex-M3) is integrated into the motherboard. It verifies the integrity of all firmware prior to boot, controls the power to rest of the Intel platform that runs the operating system

Translation: Our proprietary, impossible-to-audit, running nobody-knows-what firmware microcontroller (uC) has full authority over the boot process and execution of any system and apps running on our ORWL computer.

> ORWL’s solid state drive, an Intel SSD 540s Series, natively supports full drive encryption. The drive’s cryptographic key is generated and stored inside the secure microcontroller. The secure microcontroller provides this key to the SSD only after verifying the integrity of the system.

Translation: All the user data can be recovered by whoever has/finds a way to retrieve the key from our impossible-to-audit, impossible-to-verify uC.

> We will make available all source schematic and layout files, not just PDFs and Gerber files. We will make available all software under our control, including BIOS firmware, secure controller firmware, and key fob firmware.

Translation: At *some* point in time, we will make *select* portions of the firmware (i.e. these portions we have authored, such as maybe the logo-displaying code) available… to select partners… using yet-to-be-determined licenses.

> For those sources not under our control, in particular the data sheet for the secure controller, we will work closely with our partners to make as much information available as possible.

Translation: The datasheet for the secure uC might never be released.

> Finally the external verification process allows you to easily read the flash and verify it.

“Dear uC firmware, can you tell me if you’re a good or bad one? Just please, please, be honest, ok?”

* * *

A more perspicacious reader will surely notice that under the cover of the (debatable) humor above, I tried to conceal my disappointment with the state of vendors (not) pursuing the idea of making trustworthy personal computers.

Indeed, I could hardly point to a single vendor which would be doing anything worthwhile, and which I could wholeheartedly recommend :(

But there is a difference between taking a passive position and not progressing the art in any meaningful way (what e.g. Purism has been doing), vs. taking a step *backwards*, which is in my opinion what DesignShift attempts to do with this ORWL computing device.

The almighty, yet fully proprietary and impossible-to-audit and impossible-to-verify "secure uC" they attempt to use as a root of trust for their devices, provides for a dangerous precedent. We should not let it happen.

This ORWL's proprietary uC is not going to alleviate the problems created by Intel ME in any way. Instead it will only *add* another ME-like device, controlled by another player. In other words: another actor(s) to worry about.

Admittedly, though, the ORWL proposed physical security mechanisms, such as the board protection mesh, do indeed look interesting and potentially useful. Is there a way, then, to somehow "rescue" the ORWL device in order to benefit from these technologies? Perhaps, but as an absolute minimum, the following requirements would need to be met first:

1. The datasheet for the "secure uC" would need to be made public.

2. *All* the firmware sources, including for the uC, NFC chips, and the BIOS, would need to be published.

3. The toolchain for building all the firmware would need to be made available.

4. The firmware build process would need to be made reproducible (perhaps it is already, we don't know that).

5. The uC should expose a reliable way to dump the whole firmware through some h/w mechanism, such as a JTAG port, or at the very least allow for a reliable flashing of a new one.

Only then would it be possible to attempt to verify the security and trustworthiness of the firmware on the device.

Still, one can ask: Is this physical mesh protection really worth the effort? It might seem so at first sight. But with such a small device that costs only a few hundred dollars, another physical attack seems to be no less of a problem: the relay attack.

In the relay attack, an Evil Maid attacker replaces the *whole* target ORWL device with an identically looking one and subsequently proxies e.g. all the communication between the NFC chip and the uC to the remote (original) device, which should happily perform the challenge-response and unlock itself. The attacker is then free to penetrate the device at will.

Can ORWL provide reasonable protection against such relay attacks? Maybe. But for some reason they do not boast about it on their page, where they discuss some other attacks they attempt to address.

An inquisitive person might continue with more questions. For example, what is the exposed attack surface on the NFC stack, and what happens when the attacker successfully exploits it?

But maybe I'm being overly paranoid here. Perhaps for most users all these problems we've been discussing are just not a concern. Admittedly, that's likely the case. But perhaps such users would be better off just buying an iOS device then? Or maybe using Intel SGX-protected apps for dealing with their secrets?