microkerneldude
Random rants and pontifications by Gernot Heiser

“Trustworthy Systems Research is Done” – Are You Kidding, CSIRO?

2021/05/25
CSIRO (https://csiro.au), Australia’s national research agency, has just decided to disband the **Trustworthy Systems (https://trustworthy.systems)** (TS) team, the creators of **seL4 (https://sel4.systems)**, the worlds first operating system (OS) kernel proved correct and secure. TS is widely regarded and admired as the leaders in the use of formal methods (mathematical proof techniques) to real-world software systems, and arguably the team that put CSIRO’s Data61 on the map internationally.

Why?

Why would they cut down their shining example of research excellence, with a rare track record coming up with fundamental solutions to real problems, and taking those solutions to the real world?

One of the reasons given by CSIRO is that “**seL4 [is] now a mature technology that is ‘well supported’ outside the organisation (https://www.innovationaus.com/china-singapore-line-up-for-dumped-csiro-sel4-team/)**.”

This claim, that seL4 is a “mature technology”, i.e. no more research needed, is incredibly ignorant on so many levels. On the one hand, the group is not accidentally called “Trustworthy Systems” (and not, say, the “seL4 Research Group”). seL4 is only the starting point for achieving trustworthiness in computer systems. It’s as if over 100 years ago people said combustion engines are a solved problem once it was shown they could power a car.

Fact is that there’s plenty of fundamental research work left on seL4 itself, and there is far more research left on how to achieve real-world trustworthy computer systems. It’s not that just sprinkling a bit of seL4 fairy dust over a system will make it trustworthy. More on both points below.

In any case, it is stunning – and very depressing for the taxpayers funding CSIRO – to hear such ignorance in public statements by the national science organisation. And it’s so dramatically at odds with what is going on right now. Just as the Head of Australia’s Department of Home Affairs warns that **the threat of cyber attacks to Australia’s critical infrastructure is “immediate”, “realistic” and “credible”, and could take down the nation’s electricity network (https://www.abc.net.au/news/2021-05-24/cyber-attack-threat-critical-infrastructure-mike-pezzullo/100160894)**, CSIRO shuts down the research that specifically aims to stop such attacks (and is arguably the best approach to achieving such protection). **The mind boggles (https://twitter.com/bennoleslie/status/1396761847559163905)**.

Instead, CSIRO seems to be continuing to invest in areas like intrusion detection – which is basically an admission of defeat (trying to detect intruders means you know that you've been hacked, rather than the TS approach of preventing hacks in the first place).

Work to do: seL4

Yes, the seL4 *kernel* is mature in many ways, good enough to be deployed in real-world systems, and it is already out there in daily use in the real world, and is being designed into many more. But that doesn't mean it's "done".

Right now, seL4 solves a number of fundamental security problems, and it provides the best possible solution to these problems. In particular, it provides the strongest possible *spatial* isolation, in that it guarantees that memory cannot be accessed without explicit authorisation. It also provides strictly controlled communication between subsystems, in that two subsystems (provably) cannot communicate through system calls or memory unless explicitly authorised. And it does this with unbeaten performance. This is more than any other real-world can give you.

What seL4 cannot (yet), and no other OS can either, is to provide *temporal* isolation guarantees. This comes in two guises, the *integrity* and the *confidentiality* aspect.

Here, integrity means the ability to guarantee timeliness of real-time systems, especially *mixed-criticality systems* (MCS), where critical, high-assurance real-time tasks operate concurrently to untrusted code. seL4's new MCS model constitutes a major step in this direction, and its verification is on-going. However, it does not yet *solve* the problem. Specifically, during the verification work on the MCS kernel we came to realise that there are still issues with the model, and these issues limit its applicability to only a subset of the MCS systems we are targeting. On-going research is addressing this, leading to improvements of the model. Furthermore, we have not yet developed the formal framework for reasoning about timing guarantees on top of the MCS model. This is, of course, what is needed for making high-assurance MCS a reality.

Much more work remains on the *confidentiality* side: Here the problem is to guarantee that there is no information leakage through covert timing channels; this kind of leakage is a serious real-world problem, as demonstrated in the **Spectre attacks (https://en.wikipedia.org/wiki/Spectre_(security_vulnerability))**. Timing channels have long been put into the too-hard basket by most people; triggered by Spectre there is now a flurry of activity, mostly band-aid solutions addressing symptoms. In contrast, we are working on a principled, fundamental approach to a complete prevention of timing channels, we call this approach *time protection*, in analogy to the established memory protection. The feedback from the research community has been strong: the work has already won three best-paper awards, despite being only at the beginning.

Specifically we have worked out some basic OS mechanisms for providing time protection, and have shown that they can be effective on the right hardware, but also that present hardware is deficient. Presently, with support by the Australian Research Council and the US Air Force, we are working on proving that these mechanisms are effective on suitable hardware, and are also working with the RISC-V community on defining appropriate hardware support to allow time protection to do its job. But much more research is needed, as so far we have some basic mechanisms, that work in very restricted use cases. It's far from having an OS model that addresses the large class of systems where timing channels are a security threat.

And finally, we have not yet solved the problem on verifying seL4 for multicore platforms without sacrificing its trademark performance, in line with our motto "security is no excuse for bad performance."

So much about seL4 research being "done". It does represent the state of the art, but the state of the art is still a fair bit behind the needs of the real world.

Work to do: Scaling trustworthiness to full systems

Beyond seL4, there's the wider Trustworthy Systems agenda: **creating a *societal shift* towards mainstream adoption (https://trustworthy.systems)**, as the TS home page has been saying for years. We have made some progress here, with verification uptake increasing in academia and industry, but it's far from mainstream.

To enable this shift, the team has more concrete research goals. These include:

- Lower-cost approaches to verify the non-kernel parts of the trusted computing base, such as device drivers, file and network services, but also the actual applications. So far, verified software is still more expensive to produce than the usual buggy stuff (although life-cycle cost is probably already competitive). TS's declared aim is to produce verified software at a cost that's at par with traditionally engineered software;
- Proofs of high-level security properties of a complete system (as opposed to "just" the underlying microkernel);
- Proofs of timeliness of a complete real-time system built on seL4;
- Design of a general-purpose operating system that is as broadly applicable as Linux, but where it is possible to prove security enforcement.

These are all research challenges that remain unsolved, are of high importance for the security and safety of real-world systems and which TS is in a prime position to address.

"Research done?" Give me a break! How can anyone be so clueless?

And CSIRO says, without apparently noticing the irony, that they want to focus on "fewer, bigger things". Well, that's what made TS famous: tackling big challenges with critical mass, and solving problems no-one else could. And the above research agenda shows that we continue doing this, and we have the track record and credibility to ensure we'll continue to make good progress if people let us. If you want big results, fund the team that has a track record of delivering them!

Speaking of AI

Data61 wants to just do AI, AI, AI, and a bit of AI. Cybersecurity still appears, but is it more than lip service being paid to this National Research Priority? Cybersecurity research is supposedly still supported in two contexts, "AI for cybersecurity" and "cybersecurity for AI".

So, let’s look at the second one. AI systems are increasingly used in life- and mission-critical settings, autonomous cars are a great example. But can we trust our life on an AI system, if a hacker can bypass or influence its decision? Clearly not. In the case of an autonomous car, this means that the AI controlling the car must be protected by a secure OS. Does such a secure OS exist? We have a starting point with seL4, but to get to a real secure OS to fully protect our autonomous car, there remain a few research problems to be solved. In fact, pretty much the problems I discussed above, i.e. the very problems that define the TS research agenda.

So, if CSIRO is serious about cybersecurity for AI, then shutting down the best bet to get there seems [I’ll let you insert your own assessment here].

From → Uncategorized

Leave a Comment