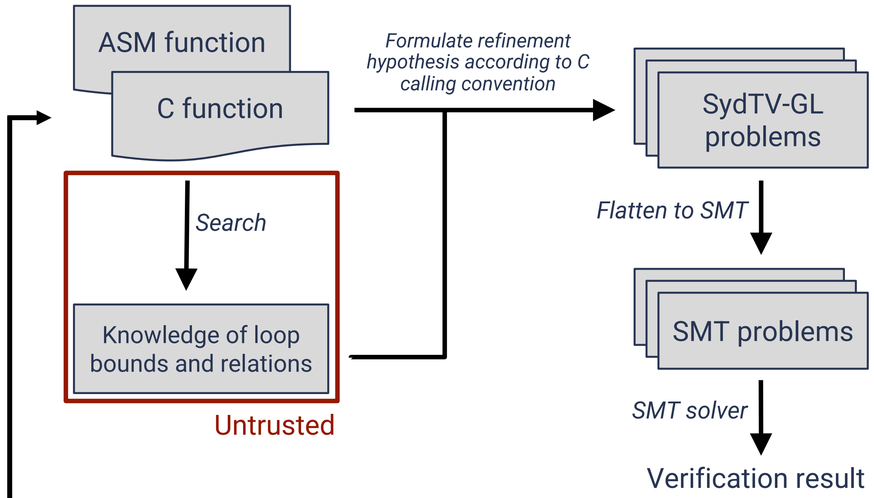
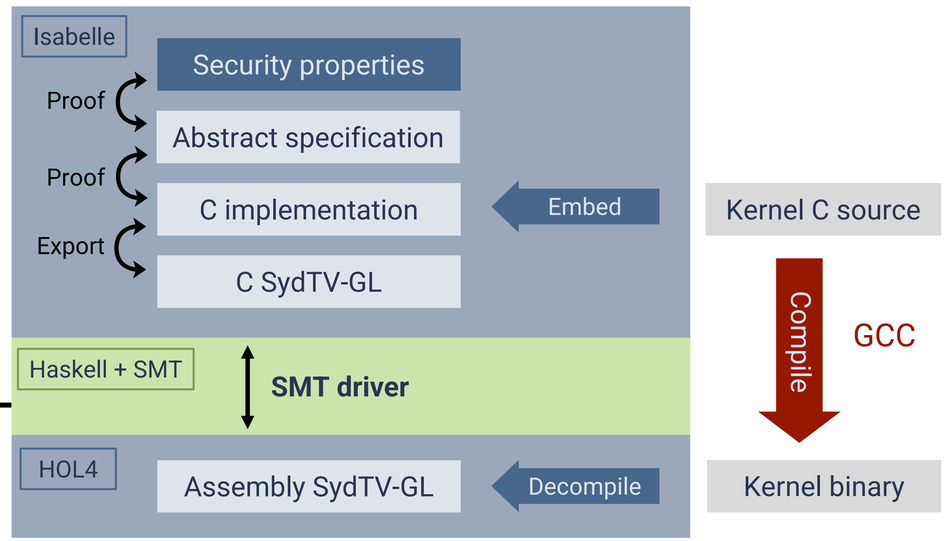
Binary Verification for seL4

Nick Spinale, Colias Group

SMT Driver



Context: seL4 Proofs



Binary verification certifies that the compiled kernel binary is a refinement of its C implementation. This completes the kernel's end-to-end assurance argument that relates abstract specification to machine code. In particular, binary verification enables us to remove the compiler from our trusted computing base.

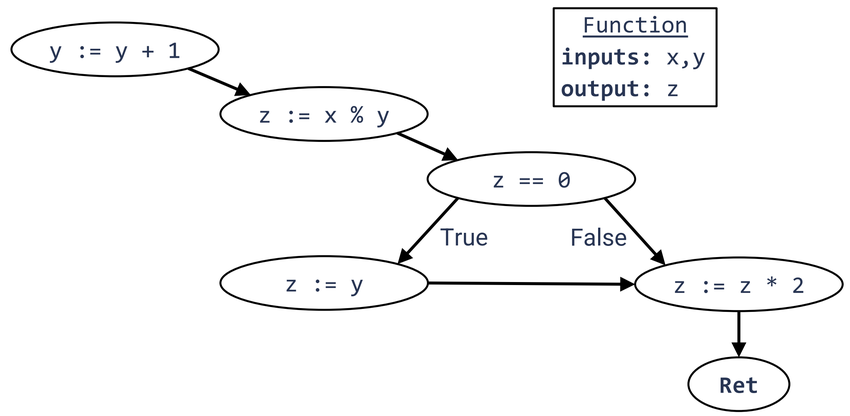
Binary verification for seL4 was originally achieved by Thomas Sewell et al. with the construction of a novel automatic binary verification toolchain[1,2]. With funding from the seL4 Foundation, we have picked up where they left off to strengthen, extend, and generalize this toolchain[3]. By **parallelizing** and **distributing** expensive computations, our in-progress new implementation of the SMT driver is **5× faster** (from 10 hours to 2 hours).

Future work

- Support for architectures beyond AArch32
- Support for even more compiler optimizations
- Generalizing the toolchain's interface to make it useful beyond seL4

SydTV-GL

Common intermediate representation

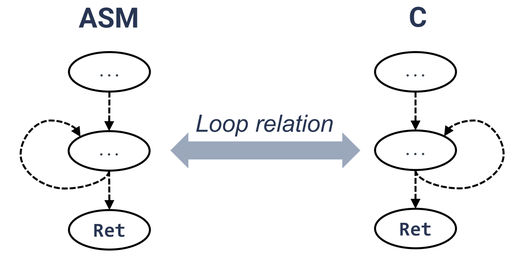


Flattened representation:

$$z_{\mathrm{Ret}} = (x_0 \,\%\, (y_0 + 1) == 0 \;?\; y_0 + 1 : x_0 \,\%\, (y_0 + 1)) * 2$$

Loop Relations

Thomas Sewell's key idea for proving refinement between unbounded loops



Definition

- Visits to loop heads correspond
- Variable values at loop head visits are either:
  - Constant or a function of visit number
  - Related to variables at corresponding visit
  - Irrelevant

Usage

1. Prove loop relation using induction
2. Treat loop as a black box assuming loop relation

References

[1]https://sel4.systems/Research/pdfs/translation-validation-verified-os-kernel.pdf
[2]https://trustworthy.systems/publications/papers/Sewell%3Aphd.pdf
[3]https://github.com/coliasgroup/seL4-binary-verification

I am at the summit! If I am not in front on my poster come and find me to talk about my work.

Nick Spinale
Colias Group