GENERAL DYNAMICS
C4 Systems

NICTA

seL4 for Dependable Systems Software

Developing dependable systems requires built-in security and safety at all levels of the system, including in the lowest-level system software: the operating system and device access software.

For truly dependable systems, the software must be trustworthy: we must be able to provide the guarantee that it behaves correctly and has the required security and safety properties. These guarantees can be provided through testing, certification, and formal verification.

Verified dependable software

Formal software verification, using machine-checked mathematical proof, provides the strongest guarantees of software properties — including correctness.

However, it is infeasible to formally verify all the code in a real system. Therefore *design for verification is crucial*. The key strategy for such design is to reduce the *trusted computing base* (TCB) — the part of the system that can break security or safety if it misbehaves. In many systems, the TCB is large. In well-designed systems, it is minimal and amenable to formal verification. The NICTA-developed *seL4 microkernel* provides such a minimal TCB.

The best way to design a system with a verifiable TCB is to split it into *critical* (trusted) and *uncritical* parts. The software that executes the system's critical functions must be trusted to have the required security and safety properties, and its execution must not be affected by any failures in the non-critical code.

Building on the seL4 microkernel

seL4 provides the secure software base that enforces separation between trusted and untrusted parts of a system.

It is a third-generation microkernel that builds on 15 years of experience with the L4 microkernel, such as small size, high performance, and policy freedom, and extends it with a mechanism to enforce security guarantees at the operating system and application levels.

seL4 is unique: it is the only operating system that has undergone formal verification, proving bug-free implementation, and enforcement of spatial isolation (data confidentiality and integrity). It is also the first protected-mode operating system with a sound timeliness analysis.

These properties allow us to reason about the security and safety of systems built on seL4.

GENERAL DYNAMICS
C4 Systems

NICTA

Verified Microkernel

What is a microkernel?

The kernel is the part of an operating system's software that operates in the hardware *privileged* mode. As such it is inherently part of a computer's TCB: it is ultimately responsible for enforcing protection and isolation, and, if faulty, can violate security and safety

A microkernel is a minimal version of an operating system kernel, providing only the bare minimum of features. Remaining operating system services sit above the microkernel, in *unprivileged* mode and rely on the microkernel to control access to the critical parts of the system.

The microkernel therefore acts as a gatekeeper: enabling the separation of *trusted* and *untrusted* functions within the system.

What is a verified microkernel?

seL4 has been comprehensively formally verified: a rigorous process to prove mathematically that its executable code, as it runs on hardware, correctly implements the behaviour allowed by the specification, and no others. Furthermore, we have proved that the specification has the desired safety and security properties (integrity and confidentiality).

One of the consequences of this formal verification is that a whole class of common programming errors cannot exist in seL4. Examples include buffer overflows or use of undefined values.

The verification was achieved at a cost that is significantly less than that of traditional high-assurance development approaches, while giving guarantees traditional approaches cannot provide.

Assumptions

In order to succeed within a useful timeframe, certain assumptions had to be made. These were chosen carefully to minimise their impact on usefulness of the verification outcome.

Some relate to the hardware and others to parts of the system, such as some assembly (low-level) code and the boot code. For a full list and explanation of their significance see http://seL4.systems. On-going work aims at removing most of these assumptions.

GENERAL DYNAMICS
C4 Systems

NICTA

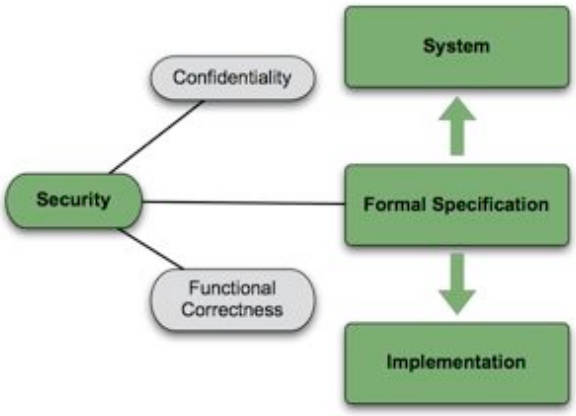
Dependable Systems

Building dependable systems

Functional correctness, as proved for the seL4 kernel, is a powerful property not available for any other operating system. Specifically it is an enabler for proving other properties that are required for true dependability.

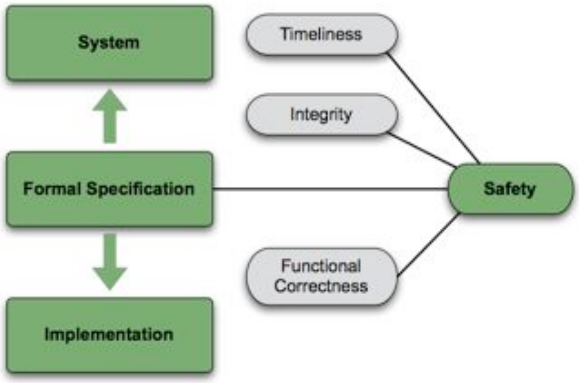
Secure systems

For security-critical applications, we must know not only that the implementation is functionally correct and that the specification is unambiguous, but also know that *critical data remains confidential*. This means that information sent between parts of the system that are meant to communicate cannot be intercepted by other (possibly malicious) parts. We have proved that for a correctly-configured system, seL4 will enforce data confidentiality.



Safety-critical systems

Safety-critical applications, on the other hand, require *data integrity*. The integrity proof guarantees that memory cannot be written by unauthorised elements in the software.



In addition, safety-critical systems often require *temporal integrity* – a guarantee that a less critical part of the system cannot affect the timeliness of a critical part.

Temporal integrity requires knowledge of the worst-case execution time (WCET) of the kernel. seL4 is the first (and still only) protected-mode kernel which has undergone a complete and sound WCET analysis.

Temporal integrity furthermore requires guarantees that a misbehaving low-criticality activity cannot affect the timeliness of a high-criticality one. We are presently working on establishing this guarantee for seL4.

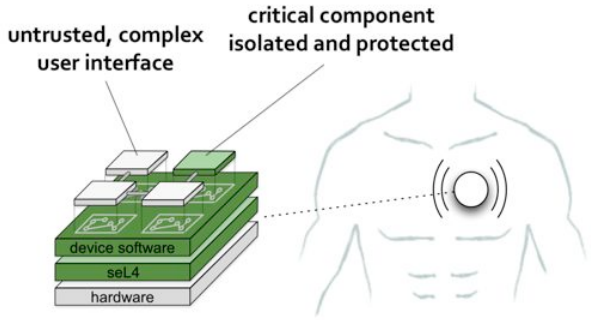
GENERAL DYNAMICS
C4 Systems

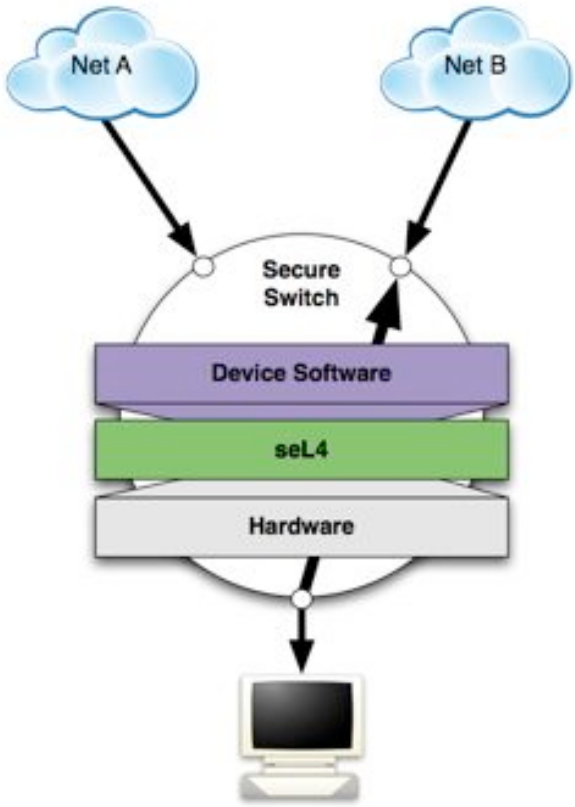
NICTA

Using seL4

How can I use it?

seL4 runs on a device's microprocessor, supporting the device's system software, as shown below in the examples of an implanted medical device and a secure network switch.





seL4 currently supports ARM and x86 processors (only ARM is presently verified). It is available as open source, see http://seL4.systems.

Track Record

seL4's predecessor, OKL4, created by the same team, has been deployed on billions of mobile and connected devices.

Further information

For further information, including details of our completed, current and planned research and technical papers, please go to: http://ssrg.nicta.com.au/projects/TS/.

Contact Details

Business Development
Jodi Steel
Tel: +61 2 9376 2123
Email: jodi.steel@nicta.com.au

Technical
Gernot Heiser
Tel: +61 2 8306 0550
Email: gernot@nicta.com.au

Australian Government

NSW Trade & Investment
Victoria
Queensland Government

QUT RMIT UNSW Federation