7th Meeting of the seL4 Technical Steering Committee

Fri, 22 Mar 2024, 18:00-19:45 AEST, by Zoom

Attendance

TSC members present:

- June Andronick (JA)
- Matthew Brecknell (MB)
- Gernot Heiser (GH)
- Gerwin Klein (GK)
- Rafal Kolanski (RK)
- Ihor Kuz (IK)
- Corey Lewis (CL)
- Kent McLeod (KM)
- Yanyan Shen (YY)

TSC members absent:

- Anna Lyons (AL)
- Kevin Elphinstone (KE)

Reviewers/Committers present:

- Indan Zupancic
- Axel Heider
- Nick Spinale

Minutes

0. **Welcome and roll call**
1. **Action items from last meeting**
    - GK: write guidelines for what should be considered a bug in Jira/issues
        - Status: not yet done.
    - IK+GK: Syncing mailing list and discourse so they can mirror each other.
        - Status: not yet done.
    - Ivan/GK: move Microkit repository to seL4 foundation
        - Status: **done**
    - Nick/GK: move rust-sel4 repository to seL4 foundation
        - Status: **done**
2. **RFC-17: IPI API**

    https://sel4.atlassian.net/browse/RFC-17

    - Overview by KM. Current implementation proposal has unicast (IPI to one core), multicast (set/bitfield of cores), and broadcast to all cores (if all bits set).

- Discussion: Implementation for now on just Arm. Ok to proceed per architecture with different implementations and tweaks, but want to agree now on an overall high-level model that they all follow.
- KM: scope for now on few cores (2-16). Discussion: current scope fine, but want model that can be conservatively extended for future many-core machines
- Indan proposes to either restrict to sending IPIs to one core at a time, or to fully expose the underlying hardware features for Arm.
- GH: for clustered multikernel likely to want to send to either one core, or to a cluster, but API model for latter still unclear (maybe a dedicated core per cluster, maybe something different).
- Discussion: providing API for unicast (to one core) is sufficient for current use cases, because sending IPIs is cheap and can be done in a loop at user space. Defer decision for multicast/broadcast/clusters until we have experience with such hardware.
- **RESOLVED (unanimously):** RFC-17 approved with following changes
  - change API to expose unicast only
  - change API to expose higher affinity bits on Arm for core addressing (fail on GICv2 and other platforms where these are not supported)

3. **RFC-16: PMU**

   https://sel4.atlassian.net/browse/RFC-16

   - Overview by GH, clarifications on RFC:
     - two supported modes of operation: global counters and events, and per-thread counters and events
     - per-thread must be virtualised/context-switched and require storage
     - global counters only need access control
     - perhaps more controversial: RFC proposes synchronous API where user says what to count and wait for interrupt. Return from kernel call only after IRQ.
   - JA: fewer security concerns in per-thread case
   - Discussion on exposing different features on different Arm implementation. Likely should restrict to architecture spec. Could make additional counters/events available per platform, but likely increased maintenance and verification cost.
   - Discussion on sync/async API, but no decision. Request for example use scenarios.
   - GK: for per-thread model, do threads measure themselves or another thread? GH: current proposal is themselves, but measuring other threads might be useful.
   - If PMU counters can be configured individually for global vs per-thread use, and PMU is per core, there is a potential for data handling mismatch in context switching when threads change core affinity. Whether this is a pure user problem is unclear (if it can violate integrity or lead to crashes, it is a kernel problem).
   - **RESOLVED (unanimously):**
     - RFC-16 needs more detail and an actual API proposal
     - detailed usage scenarios for the proposed API for config, global, and per-thread use would make the proposal clearer
   - GH agreed to update RFC and provide more detail

4. **RFC-15: Morello**

   https://sel4.atlassian.net/browse/RFC-15

   - discussion postponed

5. **RFC-14: MCS budget limit thresholds**

   https://sel4.atlassian.net/browse/RFC-14

- discussion postponed
- Indan offered to update proposed implementation by mid May to reflect current state of RFC discussion in Jira

6. **Should we remove deprecated `libsel4` functions to enable verification?**
    - GK clarified which functions
    - **RESOLVED (unanimously):** removal approved
7. **Steps for next seL4 release**
    - discussion postponed (was info only, no hold-up for release)
8. **Use GitHub for RFC process?**
    - GK: proposal to retire RFC process on JIRA and archive JIRA (leave accessible, but no longer use for new issues, incl RFC issues). Background is usability problems in JIRA and removing the need to sign up for yet another separate system for proposing RFCs.
    - **RESOLVED (unanimously):** move RFC process to GitHub, archive JIRA
    - GK+CL volunteered to adapt RFC process to GitHub and implement
9. **GitHub discussions instead of Discourse?**
    - discussion postponed

Meeting closed 19:45 AEST

Summary of Actions

- **GK**: write guidelines for what should be considered a bug in Jira/issues
- **IH/GK**: discourse + mailing list sync
- **GK+CL**: adapt RFC process to GitHub and implement
- **GH**: update RFC-16
- **Indan**: update implementation prototype for RFC-14

Acronyms

TSC Technical Steering Committee of the seL4 Foundation

Minutes prepared by JA and GK on 2024-03-27